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

D1.7: Final 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 summarize all work on this task, with a focus on work done in the second year of the project; work in the first year has already been reported in detail in Deliverable D1.3. We first situate the work with respect to the current scientific context around word embeddings, including the currently dominant contextual embedding models ELMo and BERT. We describe our recent work developing new BERT models for the EMBEDDIA project languages, show their advantage over existing multilingual models, and the ELMo models developed in the first year, on a series of extrinsic benchmarks. We then describe our new approach to intrinsic evaluation of contextual embeddings, including a new dataset, and describe the results of its use in a public shared task at SemEval 2020; and show that our new BERT models give state of the art performance. We conclude with a brief outline of some new directions.

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:	**D1.7**
Deliverable full title:	**Final context-dependent and dynamic embeddings technology**
Deliverable short title:	**Final contextual embeddings**
Document identifier:	**EMBEDDIA-D17-FinalContextualEmbeddings-T12-submitted**
Lead partner short name:	**QMUL**
Report version:	**submitted**
Report submission date:	**31/12/2020**
Dissemination level:	**PU**
Nature:	**R = Report**
Lead author(s):	**Carlos S. Armendariz (QMUL), Matej Ulčar (UL), Matthew Purver (QMUL)**
Co-author(s):	**Marko Robnik-Šikonja (UL)**
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
30/11/2020	v1.1	Carlos Armendariz, Matthew Purver (QMUL), Matej Ulčar (UL)	Initial draft.
07/12/2020	v1.2	Carlos Armendariz, Matthew Purver (QMUL), Matej Ulčar (UL)	First complete version.
14/12/2020	v1.3	Senja Pollak (JSI)	Internal review.
15/12/2020	v1.4	Mark Granroth Wilding (UH)	Internal review.
17/12/2020	v1.5	Carlos Armendariz, Matthew Purver (QMUL), Matej Ulčar (UL)	Responses to internal review.
20/12/2020	v1.6	Nada Lavrač (JSI)	Report quality checked.
22/12/2020	final	Matthew Purver (QMUL), Matej Ulčar (UL)	Final updates after quality check.
29/12/2020	submitted	Tina Anžič (JSI)	Report submitted.

Table of Contents

List of abbreviations

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

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*, and the first stage of our work in *D1.3 Initial context-dependent and dynamic embeddings technology*. To make this document self-contained, we first repeat some basic explanations and situate this work with respect to the state of the art in Section 1.1. Section 1.2 outlines the context of this deliverable within the EMBEDDIA project and presents the structure of this report.

1.1 Embeddings and the state of the art

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

1.1.2 Context-dependence

As described above, though, standard word embeddings fail to capture the fact that word meanings depend on their context. During training of an embedding, all senses of a given polysemous 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 these senses. Consequently, rare meanings of words are poorly expressed, and the resulting vectors do not offer good semantic representations.

In conceiving the EMBEDDIA project, we anticipated that performance would be significantly improved by making embedding models *context-dependent* and *dynamic*. The insight behind *context-dependent* modelling is 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 more general changes in those representations, including 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, much of the recent impressive progress in NLP has been based around 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), These have now become standard in the state of the art, giving very good performance 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. As explained in Deliverables D1.2 and D1.3, most of our work here is therefore focused on using, analysing, and improving upon these approaches, rather than developing our own from scratch.

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. Within that, the specific objective of Task T1.2 is to advance context-dependent and dynamic embeddings technology. This report describes the results of T1.2. In the first 12 months, as described in *D1.3 Initial context-dependent and dynamic embeddings technology*, the main contributions of T1.2 were:

- experiments showing the need for robust, language-specific context-dependent embeddings;
- new improved ELMo embeddings for the EMBEDDIA languages;
- extrinsic evaluation of these new ELMo embeddings on two general NLP tasks (word analogy and named entity recognition (NER));
- a new design for an intrinsic evaluation method for context-dependent embeddings, with a pilot study for a new accompanying dataset CoSimLex.

Building on those, the main contributions described here are as follows:

- new BERT models for selected combinations of EMBEDDIA languages, described in Section 2 and in the appended paper by Ulčar & Robnik-Šikonja (2020), published at the TSD 2020 conference;
- evaluation of these new BERT models, together with extended evaluation of the earlier ELMo models, presented in Section 3.
- the completed multilingual CoSimLex dataset for intrinsic evaluation, described in Section 4 and the paper by Armendariz, Purver, Ulčar, et al. (2020) published at the LREC 2020 conference, and based on a new method for crowdsourcing high-quality context-dependent judgements, described in the appended paper by Lau et al. (2020) published in the journal Transactions of the ACL;
- the results of running this evaluation method in a public SemEval shared task, Graded Word Similarity in Context (GWSC), presented in Section 4.3 and in the appended paper by Armendariz, Purver, Pollak, et al. (2020) published at the SemEval 2020 conference.
- a new method for dynamic composition of word embeddings to model how the meaning of a sentence changes as it is incrementally processed, presented in Section 5 and in the appended paper by Purver et al. (to appear).

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 D1.6 (Final cross-lingual embeddings technology) and D1.8 (Final deep neural network architectures) respectively. However while Task T1.1 focuses on methods for cross-lingual transfer, and Task T1.3 on the Deep Learning approaches themselves and making them more suitable for morphologically rich languages, Task T1.2 focuses on the dynamic and context-dependent properties of the embeddings produced. 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 six further sections. Section 2 describes the new ELMo and BERT embeddings produced in this task, including the parameters and datasets used in their development. We then present results of extrinsic evaluation for these models in Section 3, showing that they provide a new state of the art for our target languages. Section 4 describes our novel approach to intrinsic evaluation with the new CoSimLex dataset (Section 4.2), SemEval task results (Section 4.3) and the evaluation of our own new models (Section 4.4), again showing very good performance for our target languages. Section 5 then describes some new directions in our work on dynamic embeddings. We present conclusions in Section 6 where we also outline plans for further work. In addition, we list the software and models developed and released as outputs in Section 7. The appendices then include the associated published papers (Ulčar & Robnik-Šikonja, 2020; Ulčar & Robnik-Šikonja, 2020; Lau et al., 2020; Armendariz, Purver, Ulčar, et al., 2020; Armendariz, Purver, Pollak, et al., 2020; Purver et al., to appear).

2 Existing and new context-dependent embeddings

The standard state-of-the-art approach to producing context-dependent embeddings is the use of *language models* that are pre-trained on large corpora. The training task of a language model is to predict a word, when given the word's context. In some variants, the model is trained to predict the next word when given the preceding sequence; in others, it must predict a missing (or *masked*) word within an otherwise known sentence. Such a model is therefore context-dependent, as the representations it learns must generate word embedding vectors for each occurrence of a word, not just for each given word type: 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 pre-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* (Ruder et al., 2019), in which the embeddings pre-trained on a general *source* task like language modelling bring useful information about word meaning which can benefit a different *target* task. We describe existing pre-computed models and the new ones we have generated for all EMBEDDIA languages.

2.1 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, which are context dependent. 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 layers can be learned during the training of the model for a specific task. Additionally, the entire ELMo model can be fine-tuned for a specific task.

2.1.1 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. Later, ELMo models for other larger languages were trained, like Chinese and Spanish. The ELMoForManyLangs project (Che et al., 2018) 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. As reported in Deliverable D1.3 *Initial contextual embeddings*, we trained our own models, since evaluations showed the ELMoForManyLangs embeddings are significantly worse.

2.1.2 Newly developed ELMo embeddings

To obtain contextual embeddings of sufficient quality for all EMBEDDIA languages except English and Russian, we trained ELMo models for each of those languages; the resulting models have been deposited to the CLARIN repository and are publicly available (see Section 7). This work has already been reported in deliverable D1.3 *Initial contextual embeddings*, and the details of the training are presented in Appendix A. However, in addition to the work done in deliverable D1.3, we now report further evaluations of the trained ELMo models in Section 3 below.

This work is described in full in (Ulčar & Robnik-Šikonja, 2020), available as an accepted draft at the time of Deliverable D1.3, now fully published and attached here as Appendix A.

2.2 BERT

The architecture of the BERT model (Devlin et al., 2019) is composed of 12 hidden layers of Transformer encoder cells of size 768 (Vaswani et al., 2017). BERT uses two training tasks: a masked language model (MLM) attempts to predict randomly hidden tokens. A given percentage of tokens is hidden/masked in the training dataset. The second training task is 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 common words are kept as single tokens, others are split into common stems, prefixes, etc. and sometimes down to single-letter tokens.

ELMo models are typically used to produce vectors, which are then used in downstream tasks. BERT models on the other hand are generally finetuned as a whole model on a downstream task, with an added head pertaining to the desired task (sentence classification, token classification, etc.)

2.2.1 Existing BERT models

The original BERT project offers pre-trained English, Chinese, Spanish, and multilingual models. The multilingual BERT model (mBERT) 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.

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

An open source DeepPavlov library[1] offers a specific Russian BERT model, just as for ELMo. Recently, monolingual Finnish (FinBERT) (Virtanen et al., 2019), Estonian (EstBERT) (Tanvir et al., 2020) and Latvian (LVBERT) (Znotiņš & Barzdiņš, 2020) BERT models were released.

2.2.2 Newly developed BERT models

At the start of this task, there were no language-specific BERT models for EMBEDDIA languages other than the English and Russian versions mentioned above. We then trained new BERT models for EMBEDDIA languages. We decided to focus on trilingual models, featuring two similar languages and one

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

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

We have trained three trilingual models, one on Slovene, Croatian and English data (CroSloEngual BERT), one on Estonian, Finnish and English (FinEst BERT), and one on Latvian, Lithuanian and English (LitLat BERT). The models are now publicly available via the popular Huggingface library and for individual download from CLARIN (see Section 7).

For each model we combined deduplicated corpora from all three languages. The corpora used to train our BERT models are described in Section 2.3. The corpora are mostly the same as those used to train ELMo models, but we added additional corpora for training LitLat BERT. To Latvian corpora we added Saeima corpus (Dargis et al., 2018), Latvian part of DGT-UD corpus[2] and Latvian articles from Ekspress Meedia. Adding these corpora raised the total size of Latvian data from 0.27 billion tokens to 0.53 billion tokens after deduplication.

FinEst BERT and CroSloEngual BERT were trained on BERT-base architecture. We used bert-vocab-builder[3] to produce wordpiece vocabularies (composed of subword tokens) from the given corpora. The created wordpiece vocabularies contain 74,986 tokens for FinEst and 49,601 tokens for CroSloEngual model. The training dataset is a masked corpus. We randomly masked 15% of the tokens in the corpus and repeated the process 5 times, each time with different 15% of the tokens being masked. The dataset is thus five times larger than the original corpora. On this data we trained our BERT models for about 40 epochs, which is approximately how much the multilingual BERT was trained for. The details of the training are described in Appendix B.

LitLat BERT is based on the RoBERTa architecture. We opted for the RoBERTa approach because it has since proven itself as more robust and better performing than BERT. It also offered two practical benefits over original BERT approach. By dropping the next-sentence prediction training task, corpora shuffled on the sentence level can easily be used. The second benefit is that it allows for training on multiple GPUs out of the box, while BERT can only be trained on a single GPU, unless complex workarounds are implemented.

We split the Lithuanian, Latvian and English corpora into three sets, train, eval and test. Train dataset contains 99% of all the corpora, the other two sets contain 0.5% each. We used sentencepiece[4] to produce subword byte-pair-encodings (BPE) from a given train dataset. The created subword vocabulary contains 84,200 tokens. We have trained the model for 38 epochs[5], with maximum sequence length of 512 tokens. Just like with FinEst and CroSloEngual BERT, we randomly masked 15% of the tokens during the training.

The FinEst BERT and CroSloEngual BERT work in this section is described in full in (Ulčar & Robnik-Šikonja, 2020), attached here as Appendix B.

2.3 Training corpora

We reported the corpora used to train ELMo models for all the EMBEDDIA languages in Deliverable D1.3 *Initial contextual embeddings*. The corpora we used for training new BERT models is shown in Table 1 for each language separately. The English dataset used in our BERT models is the same in all of the models. The corpora is mostly the same as reported in deliverable D1.3. However, we added some additional corpora. Some corpora are available online under permissive licences, some are available only for research, and some were provided by the project partners. Further details can be found in Deliverable D1.1 *Datasets, benchmarks and evaluation metrics for cross-lingual word embeddings*.

[2] http://hdl.handle.net/11356/1197
[3] https://github.com/kwonmha/bert-vocab-builder
[4] https://github.com/google/sentencepiece
[5] the model has not been fully trained yet

Table 1: The training corpora for BERT models and their total size (in billions of tokens) per language.

Language	Corpora	Size
Croatian	hrWaC 2.1, Riznica, Styria articles	1.95
English	1 Billion Word Benchmark	0.8
Estonian	CoNLL 2017, Ekspress Meedia articles	0.68
Finnish	STT articles, CoNLL 2017, Ylilauta downloadable version	0.92
Latvian	CoNLL 2017, DGT-UD, Saeima, Ekspress Meedia articles	0.53
Lithuanian	Wikipedia 2018, DGT-UD, LtTenTen14	1.30
Slovene	Gigafida 2.0	1.26

3 Extrinsic evaluation

In Deliverable D1.3 *Initial contextual embeddings*, we evaluated our new ELMo models on two tasks: word analogy and named entity recognition (NER). In this report, while our ELMo models are the same as reported in D1.3, we improve their evaluation on the NER task by improving the architecture of the named entity recogniser and by comparing against one more reference model. We also evaluate the ELMo models on a new task: dependency parsing. We then evaluate our new BERT models on two evaluation tasks: named entity recognition (NER) and dependency parsing (DP), and compared their performance with existing BERT models.

3.1 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 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). Label frequencies in the datasets for each language are shown in Table 2.

Table 2: 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	10241	7445	11216	0.057
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

3.1.1 ELMo

We embedded the text in the datasets with ELMo models to produce three vectors for each token. We used the three ELMo vectors as the input of our recognition model. We first calculated a weighted average of them, where the weights were learned during the training. This was followed by two bidirectional LSTM layers with 2048 LSTM cells. The final layer of our model was a time-distributed softmax layer with 4 neurons. Details about training the NER models are described in Appendix A.

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). We compared the results of our ELMo emeddings with embeddings generated by ELMoForManyLangs embeddings in Table 3. In the same table we also present the results achieved by using non-contextual fastText embeddings. Both ELMo variants significantly outperform fastText embeddings on the task. On Latvian, ELMoForManyLangs and EMBEDDIA ELMo embeddings perform equally well. On other languages, EMBEDDIA ELMo improve results over ELMoForManyLangs baseline.

Table 3: The results of NER evaluation task. The scores are macro average F_1 scores of the three named entity classes, excluding score for class "Other". The columns show fastText and ELMoForManyLangs (EFML) as baselines, and our new EMBEDDIA ELMo embeddings.

Language	fastText	EFML	EMBEDDIA
Croatian	0.62	0.73	**0.82**
Estonian	0.79	0.89	**0.91**
Finnish	0.76	0.88	**0.92**
Latvian	0.62	**0.83**	**0.83**
Lithuanian	0.44	N/A	**0.74**
Slovenian	0.63	0.82	**0.85**
Swedish	0.75	0.85	**0.88**

3.1.2 BERT

We evaluated EMBEDDIA BERT models (CroSloEngual BERT and FinEst BERT) on the NER task by finetuning the entire models with an added token classification head for this task. We used transformers library by Huggingface (Wolf et al., 2020) to finetune the models on NER datasets for 3 epochs. We compare our models with existing multilingual BERT models: multilingual BERT (mBERT), XLM-RoBERTa, and with existing monolingual BERT models on appropriate languages: Finnish (FinBERT), Estonian (EstBERT) and Latvian (LVBERT).

The results are presented in Table 4. CroSloEngual BERT outperforms all other evaluated models on Croatian, Slovenian and English. FinEst BERT performs comparably to the FinBERT and beats all other models on Finnish. On Estonian FinEst BERT outperforms all other evaluated models. LitLat BERT outpeforms mBERT and XLM-R on Lithuanian. On Latvian, LitLat BERT achieves similar results as XLM-R, with both of them beating mBERT. Monolingual EstBERT and LVBERT do not perform on par with other models on their respective languages. Surprisingly, FinBERT performed better on Estonian than EstBERT (0.876 and 0.872 macro F_1 scores, respectively).

Table 4: The results of NER evaluation task for various BERT models. The scores are macro average F_1 scores of the three NE classes. NER models were fine-tuned from each of the BERT models. “MONO” represents monolingual models (FinBERT for Finnish, EstBERT for Estonian, LVBERT for Latvian). We compare our models CroSloEngual BERT (CSE), FinEst BERT and LitLat BERT against baseline mBERT, XLM-R and monolingual (MONO) models.

				EMBEDDIA		
Language	mBERT	XLM-R	MONO	CSE	FinEst	LitLat
Croatian	0.790	0.817	-	**0.884**	-	-
English	0.939	0.937	-	0.944	**0.945**	0.942
Estonian	0.898	0.908	0.872	-	**0.927**	-
Finnish	0.933	0.930	**0.961**	-	0.957	-
Latvian	0.830	0.865	0.797	-	-	**0.867**
Lithuanian	0.797	0.817	-	-	-	**0.852**
Slovenian	0.897	0.914	-	**0.920**	-	-

3.2 Dependency parsing

Dependency parsing task (DP) attempts to predict the dependency tree structure of a given sentence. The words in said sentence are arranged into the tree, based on the syntactical relations between them. Each node in the tree represents one word and has at most one parent. The task also attempts to classify these relations with an appropriate label from a given set of labels (Jurafsky & Martin, 2009).

We trained dependency parsers on universal dependencies (UD) treebank datasets, version 2.3 (Nivre et al., 2018). Specifically, we used the following datasets: SET for Croatian, EWT for English, EDT for Estonian, TDT for Finnish, LVTB for Latvian, ALKSNIS for Lithuanian, GSD for Russian, SSJ for Slovenian, and Talbanken for Swedish.

We present the results of dependency parsing task as unlabeled attachement score (UAS) and labeled attachment score (LAS). The UAS score is defined as the proportion of tokens that are assigned the correct parent in the tree (or are correctly identified as roots), while the LAS score is the proportion of tokens that are assigned the correct head as well as the correct dependency relation label.

3.2.1 ELMo

To train dependency parsers using ELMo embeddings, we used SuPar tool by Yu Zhang.[6] SuPar is based on the deep biaffine attention (Dozat & Manning, 2017). We modified the SuPar tool to accept ELMo embeddings on the input; specifically, we used the concatenation of the three ELMo vectors. The modified code has been made publicly available (see Section 7). We trained the parser for 10 epochs for each language, using separately EMBEDDIA ELMo embeddings and ELMoForManyLangs embeddings.

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

We present the results in Table 5. EMBEDDIA ELMo embeddings perform better than ELMoForManyLangs on all languages, for both metrics. We add the comparison between the original English ELMo and English ELmoForManyLangs embeddings. The difference in performance between the models is smallest on Latvian, English and Swedish. The relatively small improvement of our ELMo on Latvian can be explained by the usage of a small training corpus. Neither English nor Swedish are morphologically rich languages, which could explain the smaller improvement on those two languages.

Table 5: The ELMo embeddings quality measured on the dependency parsing task. Results are given as UAS and LAS for EMBEDDIA ELMo and ELMoForManyLangs. For English, the original ELMo model is shown instead of EMBEDDIA ELMo. There is no Lithuanian ELMoForManyLangs model.

	ELMoForManyLangs		EMBEDDIA ELMo	
Language	UAS	LAS	UAS	LAS
English	0.903	0.863	**0.905**	**0.872**
Slovenian	0.856	0.777	**0.937**	**0.914**
Croatian	0.882	0.795	**0.917**	**0.858**
Finnish	0.883	0.834	**0.908**	**0.869**
Estonian	0.812	0.725	**0.895**	**0.855**
Latvian	0.872	0.808	**0.889**	**0.828**
Lithuanian	-	-	**0.551**	**0.244**
Swedish	0.880	0.831	**0.897**	**0.851**

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

3.2.2 BERT

Similarly to Section 3.2.1, we next used the SuPar tool to train a BERT-based dependency parser. We extracted the vectors from the last four BERT layers and fed them to the input of the dependency parsing model. We used the weighted average of the four vectors and trained for 10 epochs for each language, using EMBEDDIA CroSloEngual, FinEst and LitLat BERT models. We compare the results with vectors from multilingual BERT (mBERT) and XLM-RoBERTa (XLM-R) as reference.

Table 6: The BERT embeddings quality measured on the dependency parsing task, using vectors extracted from last four layers of BERT. Results are given as UAS and LAS.

	mBERT		XLM-R		EMBEDDIA CroSloEngual		EMBEDDIA FinEst		EMBEDDIA LitLat	
Language	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS
Croatian	0.922	0.861	**0.931**	**0.875**	0.925	0.861	-	-	-	-
Slovenian	0.927	0.898	**0.947**	**0.927**	0.930	0.896	-	-	-	-
English	0.915	0.878	**0.929**	**0.895**	0.918	0.881	0.907	0.862	0.758	0.658
Estonian	0.849	0.791	**0.893**	**0.849**	-	-	0.843	0.762	-	-
Finnish	0.871	0.810	**0.915**	**0.875**	-	-	0.862	0.777	-	-
Latvian	0.841	0.758	**0.897**	**0.836**	-	-	-	-	0.547	0.333
Lithuanian	**0.569**	**0.263**	0.522	0.228	-	-	-	-	0.250	0.052

CroSloEngual BERT performs on par with mBERT, while FinEst BERT performs slightly worse than mBERT. LitLat model performs badly on all languages, which can be partly explained by the fact that the model has not been fully trained yet. However, XLM-RoBERTa significantly outperforms all of the other BERT models on this task, except on Lithuanian. Full results are displayed in Table 6. Our own models achieved very poor results on Lithuanian with all ELMo and BERT embeddings, which indicates a low quality dataset, so the results on this language might not be the most representative.

However, while this method allows direct comparison between our BERT models and the ELMo models from Section 3.2.1, it is not the optimum way to deploy BERT. BERT models are usually fully fine-tuned on an end-task, and show better results that way than by extracting vectors from a pre-trained BERT model and using them directly to train a classifier. We used the Udify tool (Kondratyuk & Straka, 2019) to train the dependency parsing classifier by fine-tuning the entire BERT models. We trained the dependency parser for 80 epochs on the treebank data. We kept the tool parameters at default values, except for "warmup_steps" and "start_step" values, which we changed to equal the number of training batches in one epoch. The tool does not support finetuning RoBERTa-type models, so we were not able to evaluate XLM-R and LitLat BERT models on this task.

Table 7: The BERT embeddings quality measured on the dependency parsing task by fine-tuning the whole BERT models. Results are given as UAS and LAS.

	mBERT		EMBEDDIA CroSloEngual		EMBEDDIA FinEst	
Language	UAS	LAS	UAS	LAS	UAS	LAS
Croatian	0.930	0.891	**0.940**	**0.903**	-	-
Slovenian	0.938	0.922	**0.957**	**0.947**	-	-
English	0.917	0.894	**0.922**	**0.899**	0.918	0.895
Estonian	0.880	0.848	-	-	**0.909**	**0.882**
Finnish	0.898	0.867	-	-	**0.933**	**0.915**

Results are shown in Table 7: CroSloEngual BERT now shows improvement over mBERT on all three languages, with the highest improvement on Slovenian and only a marginal improvement on English. FinEst BERT outperforms mBERT on Estonian and Finnish, with the biggest margin being on the Finnish data. FinEst BERT and mBERT perform equally well on English data.

Relative to multilingual BERT performance, EMBEDDIA BERT models perform significantly better when fine-tuning the entire model than when extracting vectors from a pre-trained model.

The BERT results in this section are described in full in (Ulčar & Robnik-Šikonja, 2020), attached here as Appendix B.

4 Intrinsic evaluation

In contrast with *extrinsic* evaluation, in which the embeddings are evaluated by their performance in specific tasks, *intrinsic* evaluation focuses on properties of the embeddings themselves. Intrinsic evaluation of word embeddings is usually performed via word similarity and relatedness tests, in which the distance between two embeddings produced by the model is compared against datasets containing ratings by human annotators. Exact metrics vary: for example, earlier datasets like WordSim-353 (Finkelstein et al., 2001) and MEN (Bruni et al., 2014) didn't make a clear distinction between relatedness and similarity, whereas the later SimLex-999 (Hill et al., 2015) made a point of making a clear distinction between the two and focusing on similarity ratings. However, the main problem with these datasets, and most intrinsic evaluation methods for embeddings to date, is that they do not take context into account, but are based only on properties of words when seen in isolation: they are thus not suited to evaluating the context-dependent embeddings of interest here. Some recent work has introduced context-dependence, 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 word *senses*; as such, it is not 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. The work presented in this section attempts to fill this gap.

Deliverable D1.3 (Initial context-dependent and dynamic embeddings) introduced the concept behind this approach, and described initial pilot studies and a proposal for an evaluation task. Here, we present the final approach, complete with multi-lingual public dataset and application both as a public shared task and as an evaluation for the models described in Section 2 above.

4.1 Approach

Our new approach is based on the approach of SimLex-999 (Hill et al., 2015), in which the evaluation focuses on the embedding model's ability to predict human judgements of similarity between word pairs; the novelty is that we move this to a context-dependent setting in which: (a) the pairs are presented in context (so that the effect of context on similarity must be predicted); and (b) each pair is presented in more than one different context (so that the change between contexts must be predicted).

Figure 1 shows an example. Systems are presented with a paragraph of text, and must predict human judgements of similarity of two target words contained therein. The same target word pair is presented in two different contexts, and thus paired with two corresponding different gold standard judgements. A successful model must therefore accurately reflect human perceptions of the effect of context. In contrast to existing approaches (Huang et al., 2012; Pilehvar & Camacho-Collados, 2018), in which systems must only predict whether words have the same sense or not in different contexts, models can now be evaluated on their ability to predict the graded, numerical measures of human similarity perception; the changes that the change in context induces; or both.

4.2 Dataset: CoSimLex

In order to support this new approach to intrinsic evaluation of contextual embeddings we created a new dataset, CoSimLex (Armendariz, Purver, Ulčar, et al., 2020). This dataset is based on pairs of words

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

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

from SimLex-999 (Hill et al., 2015) to allow comparison with the context-independent case. For Croatian and Finnish we use existing translations of SimLex-999 (Mrkšić et al., 2017; Venekoski & Vankka, 2017; Kittask, 2019). In the case of Slovene, we produced our own new translation following Mrkšić et al. (2017)'s methodology for Croatian. This Slovenian translation has been made publicly available via CLARIN.[7]

CoSimLex consists of 340 word pairs in English, 112 in Croatian, 111 in Slovene and 24 in Finnish. Each pair is rated within two different contexts, giving a total of 1174 scores of contextual similarity. Each line of CoSimLex is made of a pair of words; two different contexts extracted from Wikipedia in which these two words appear; two scores of similarity, each one related to one of the contexts, calculated as the mean of annotator ratings for that context; two scores of standard deviation; the p-value given by applying the Mann-Whitney U test to the two score distributions; and the four inflected forms of the words exactly as they appear in the contexts (including case; note that in the morphologically rich languages, many inflections are possible). To the best of our knowledge, this is the first reasonably sized dataset in which differences in contextual similarity between two words are supported with a test of statistical significance. Figure 1 shows an example from the English dataset.

4.2.1 Context selection

One main challenge was selecting two suitable real contexts in which to present each word pair. Contexts were extracted from each language's Wikipedia; they are made of three consecutive sentences, and contain the pair of words, appearing only once each. To maximise the likely difference in ratings of similarity between contexts, we developed a two-stage process based on existing ELMo and BERT models and human judges. In the first step, we used ELMo and BERT models to rate the similarity between the target words in context, and selected candidate contexts with the highest and lowest scores, together with some randomly selected. The second step was performed by expert human annotators, one per language, who were asked to select the two that, in their opinion, maximised the contrast in similarity.

4.2.2 Annotation and pre-processing

The other major challenge was developing a process to collect reliable human judgements of similarity in context. We based this on the instructions used for SimLex-999 (Hill et al., 2015), and for English adopted their use of crowd-sourcing via *Amazon Mechanical Turk*, while for the less-resourced languages where few crowd-source workers are available, we recruited annotators directly.

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

The process of annotating ratings in context is relatively complex, and over a number of pilot studies we discovered that obtaining good inter-rater agreement (IRA) depended on careful design of the interface and process: in particular, it is crucial to maximise the annotators' engagement with the context paragraphs (particularly in the crowd-sourced setting where annotators are particularly keen to minimise the time spent on any task). Through iterative design and pilot testing, we arrived at a two-step annotation process: in the first step, annotators must read the context paragraph and perform a small task which ensures that it has been properly read; only after completing that do they see the target words highlighted in bold and are asked to rate their similarity. Reliability of annotation was also ensured by an adapted version of SimLex-999's post-processing method, which includes rating calibration and the filtering of annotators with very low correlation to the rest.

As we can see in Table 8, the resulting CoSimLex datasets in the different languages show good IRA correlation scores, very close to those of SimLex-999 (ρ = 0.77 vs ρ = 0.78 in English); show a high percentage of statistically significant differences between contexts (62% of pairs overall); and show very similar average change between contexts, even for the highly inflected languages. The CoSimLex dataset is now available in the public repository CLARIN.[8]

Table 8: Similarity, standard deviation, Spearman's ρ and change are average values. The two rightmost columns denote the proportion of pairs whose differences of scores with the original values are statistically significant at p-value < 0.1 and p-value < 0.05.

Dataset	#pairs	Sim	StDev	Spearman's ρ	Change (Abs)	$p < 0.1$	$p < 0.05$
SimLex-999	999	4.56	1.27	0.78	-	-	-
English CoSimLex	340	5.54	2.24	0.77	2.16	65%	61%
Croatian CoSimLex	112	4.39	2.23	0.76	2.32	65%	54%
Slovene CoSimLex	111	4.90	2.17	0.77	1.96	59%	46%
Finnish CoSimLex	24	4.08	2.16	0.81	1.75	33%	29%

This work is described in full in two papers. Our method for crowdsourcing contextual judgements is described in (Lau et al., 2020), attached here as Appendix C. The CoSimLex dataset itelf is described in (Armendariz, Purver, Ulčar, et al., 2020), available as an accepted draft at the time of Deliverable D1.3, now fully published and attached here as Appendix D.

4.3 SemEval2020 Task 3: Graded Word Similarity in Context

Our approach and dataset were put into practice as a public shared task, named *Graded Word Similarity in Context (GWSC)*.[9] This was run 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.[10] The evaluation phase ran during February and March 2020; we received 15 submissions from teams all over the world. A more detailed look at the outcomes can be found in our task description paper (Armendariz, Purver, Pollak, et al., 2020) which was presented in December at the 28th International Conference of Computational Linguistics (COLING'2020[11]); 11 participating teams also presented papers about their submissions to our competition.

4.3.1 Subtasks and metrics

The task was multi-lingual, using the CoSimLex datasets in the four EMBEDDIA languages mentioned above (English, Slovenian, Croatian and Finnish). System performance was evaluated using two independent metrics, which measure different aspects of prediction quality:

[8] https://www.clarin.si/repository/xmlui/handle/11356/1308
[9] https://competitions.codalab.org/competitions/20905
[10] https://semeval.github.io/
[11] https://coling2020.org/

Subtask 1 - Predicting Changes: The first aspect is the ability of a system to predict the *change in similarity ratings between the two contexts* for each word pair. This is evaluated via the correlation between the changes predicted by the system and those derived from human ratings. We use the uncentered Pearson correlation: this gives a measure of accuracy of predicting relative magnitude of changes, and allows for differences in scaling while maintaining the effect of direction of change (the standard centered correlation normalizes on the mean, so could give high values even when a system predicts changes in the wrong direction, but with a similar distribution over examples).

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

Subtask 2 - Predicting Ratings: The second aspect is the ability to predict the absolute similarity rating for each word pair in each context. This was evaluated using the harmonic mean of the Pearson and the Spearman correlation with gold-standard judgements, following the example of SemEval2017 Task2: Multilingual and cross-lingual semantic word similarity (Camacho-Collados et al., 2017).

Teams were able to submit different systems for each subtask and language pair. The baselines were based on the multilingual version of the popular context-dependent model BERT. Additionally, we provided the results achieved by the ELMo models trained by the EMBEDDIA team.

4.3.2 Task submissions and rankings

The task received a total of 14 submissions for the first subtask and 15 submissions for the second. From those, 11 teams submitted system description papers for review. As can be seen in tables 9 and 10, systems beat the baselines by significant margins, but few did well in more than one language or subtask.

Table 9: Subtask 1 Final Ranking: The values are calculated as the Pearson Uncentered Correlation between the system's scores and the average human annotation. It represents the system's ability to predict the change in perception produced by the contexts. Since different annotators looked at each context, human performance couldn't be calculated for this subtask. JUSTMasters and UZH are not part of the official ranking since they were able to optimise their systems with more than the competition's limit of 9 submissions.

SUBTASK 1							
English		Croatian		Slovene		Finnish	
Ferryman	0.774	BabelEnconding	0.74	Hitachi	0.654	will_go	0.772
will_go	0.768	Hitachi	0.681	BRUMS	0.648	Ferryman	0.745
MultiSem	0.76	BRUMS	0.664	BabelEnconding	0.646	BabelEnconding	0.726
LMMS	0.754	Ferryman	0.634	CiTIUS-NLP	0.624	BRUMS	0.671
BRUMS	0.754	LMMS	0.616	Ferryman	0.606	CiTIUS-NLP	0.671
Hitachi	0.749	will_go	0.597	will_go	0.603	MultiSem	0.593
BabelEnconding	0.73	CiTIUS-NLP	0.587	LMMS	0.56	Hitachi	0.574
CiTIUS-NLP	0.721	MineriaUNAM	0.374	MineriaUNAM	0.328	MineriaUNAM	0.389
MineriaUNAM	0.544	MultiSem	-	MultiSem	-	LMMS	0.36
JUSTMasters	0.738		0.44		0.512		0.546
UZH	0.765		-		-		-
mBERT_uncased	0.713		0.587		0.603		0.671
ELMo	0.570		0.662		0.452		0.550

A lot of the systems submitted were based on (or at least made use of) models inspired by BERT or other Transformer based models. However we saw a variety of ideas on how best to leverage this model's power. Some teams found ways to make use of external knowledge: **LMMS** and **AlexU-AUX-BERT** created sense embeddings using WordNet (Miller, 1995) as a guide. The approach worked very well for English, but proved difficult to apply to other languages. **Ferryman**'s model finished first in the

Table 10: Subtask 2 Final Ranking: The values are calculated as the harmonic mean of the Spearman and Pearson correlation between the system's scores and the average human annotation. It represents the system's ability to predict contextual human perception of similarity. Human performance is the average value when comparing each annotator against the average of the rest. JUSTMasters is not part of the official ranking since they were able to optimise their system with more than the competition's limit of 9 submissions.

SUBTASK 2							
English		Croatian		Slovene		Finnish	
MineriaUNAM	0.723	BabelEnconding	0.658	BabelEnconding	0.579	BRUMS	0.645
LMMS	0.72	Hitachi	0.616	BRUMS	0.573	BabelEnconding	0.611
AlexU-Aux-Bert	0.719	MineriaUNAM	0.613	CiTIUS-NLP	0.538	MineriaUNAM	0.597
MultiSem	0.718	LMMS	0.565	will_go	0.516	MultiSem	0.492
BRUMS	0.715	BRUMS	0.545	AlexU-Aux-Bert	0.516	Ferryman	0.357
will_go	0.695	CiTIUS-NLP	0.496	Hitachi	0.514	LMMS	0.354
Hitachi	0.695	AlexU-Aux-Bert	0.402	MineriaUNAM	0.487	will_go	0.35
CiTIUS-NLP	0.687	will_go	0.402	LMMS	0.483	Hitachi	0.335
BabelEnconding	0.634	Ferryman	0.397	Ferryman	0.345	CiTIUS-NLP	0.289
Ferryman	0.437	MultiSem	-	MultiSem	-	AlexU-Aux-Bert	0.289
JUSTMasters	0.725		0.443		0.44		0.68
mBERT_uncased	0.573		0.402		0.516		0.289
ELMo	0.510		0.529		0.407		0.516
Human	0.77		0.76		0.77		0.81

English subtask 1 by feeding the TF-IDF score of the words in the contexts to their BERT inspired model during training.

Some teams worked on ways to increase the data available to the model. The **MultiSem** team created five new datasets in order to fine-tune their BERT models. Most of them were automatically generated from previous datasets to increase contextual influence. With a very multiligual approach, **BabelEnconding** translated the contexts and target words to many languages and fed them to their models. They reasoned that, in addition to increasing the amount of data available, the translation could help with things like word sense disambiguation when two senses are actually a different word in some of the languages. It worked especially well for the less resourced languages.

Another group of submissions focused on testing a variety of models and parameters and combining them in different ways, often to create stacked embeddings. The **BRUMS**,**Hitachi** and **JUSTMasters** teams fall in this category. Finally **MineriaUNAM** used K-Means inspired centroids to modify the original SimLex-999 non contextualised similarity scores. It delivered great results for the English subtask 2, which they won, but it would be great to see the same ideas applied in a way that doesn't depend on the original human annotation.

Many of the improvements in performance seen in these submissions were dependent on additional resources like WordNet, customised datasets and previous human annotations. Sadly this makes these approaches very difficult to apply to less resourced languages. A notable exception was the submission by the **BabelEnconding** team which, in addition to the great performance of the system, led us to award them with the best paper of SemEval2020 Task 3.

This work is described in full in (Armendariz, Purver, Pollak, et al., 2020), attached here as Appendix E.

4.4 Intrinsic evaluation of EMBEDDIA embeddings

As administrators of the task, we did not take part in the competition; but can now use the same method to evaluate and compare our new EMBEDDIA models described in Sections 2 and 3 above. As we can see in Tables 11 and 12, our ELMo models outperformed the existing "ELMo for many languages" (Che et al., 2018) in every subtask and language. The difference is more significant for Croatian and Slovene.

As we saw with the SemEval task baselines, the Transformer-based BERT models do significantly better than ELMo. Our BERT models, trained in a small selection of related languages, did much better than the existing multilingual BERT (mBERT), which was trained in a large number of different languages. Croatian and Slovene were again the two languages that benefited the most.

Comparing with the submissions for the SemEval2020 task, our BERT models would have finished first in both subtasks for Slovene, second in both for Croatian, and fourth in both for Finnish. It is important to note that our models are trained as general language model embeddings, rather than being specially optimised for this specific task like some of the SemEval task submissions. While some of the SemEval approaches do not seem very well suited to less-resourced languages, as discussed above, others might be more appropriate (e.g. BabelEnconding's use of simultaneous multiple-language embeddings) - it will be interesting to see if techniques like this can help improve performance in other NLP tasks.

Table 11: Subtask 1. Uncentered Spearman Correlation.

	Model	English	Croatian	Slovene	Finnish
ELMo models	ELMoForManyLangs	0.556	0.520	0.467	0.403
	EMBEDDIA ELMo	**0.570**	**0.662**	**0.550**	**0.452**
BERT models	mBERT_uncased	0.713	0.587	0.603	0.671
	EMBEDDIA CroSloEng BERT	**0.719**	**0.715**	**0.673**	-
	EMBEDDIA FinEst BERT	0.692	-	-	**0.672**

Table 12: Subtask 2. Harmonic mean of the Spearman and Pearson correlations.

	Model	English	Croatian	Slovene	Finnish
ELMo models	ELMoForManyLangs	0.449	0.433	0.328	0.403
	EMBEDDIA ELMo	**0.510**	**0.529**	**0.516**	**0.407**
BERT models	mBERT_uncased	0.573	0.443	0.516	0.289
	EMBEDDIA CroSloEng BERT	**0.601**	**0.642**	**0.589**	-
	EMBEDDIA FinEst BERT	0.591	-	-	**0.533**

5 New directions in dynamic embeddings

As outlined in Deliverable D1.3, the strong performance of contextual embedding models such as BERT, which can be pre-trained on general language modelling tasks on very large amounts of text and then fine-tuned for specific tasks, have quickly made them the predominant state-of-the-art approach in most NLP tasks. However, the notion of context-dependence that they incorporate is a restricted one of the context of a word within a surrounding word sequence (usually a sentence), and this leaves open some questions about how to model other notions of context. In this section, we briefly outline some of our work into other directions.

5.1 Short-term context: incremental parsing

Models such as ELMo and BERT assume that sentential context is always available: a word is represented and modelled within the context of an entire surrounding word sequence, either by bidirectional sequence modelling (ELMo) or omnidirectional connection (BERT). A rather different notion of context is required for models of human language processing, particularly when viewed in the context of spoken language (as is required in dialogue systems): humans process language incrementally, left-to-right, with words understood and produced in the evolving context of the words heard or spoken so far. ELMo/BERT-style approaches are therefore not suitable as models, and approaches that might suit a left-to-right setting (e.g. unidirectional LSTMs) are not generally designed to produce representations which match the incrementality of human perception.

Our initial work in this direction aims to develop a model in which embeddings for an incrementally developing sentence can be calculated by a compositional process which combines the static embeddings of the constituent words in a left-to-right way, producing a suitable representation at each stage. Although currently implemented only for simple sentences, the composition depends on semantic predicate-argument relations and thus can be transferred in principle to any language into which the grammar can be transferred. Results show that its ability to incrementally disambiguate word senses outperforms that of the standard approach used to combine static word embeddings (simple addition) - see Figure 2. In future work we plan to investigate how this approach can be integrated with the standard context-dependence of ELMo/BERT-style models.

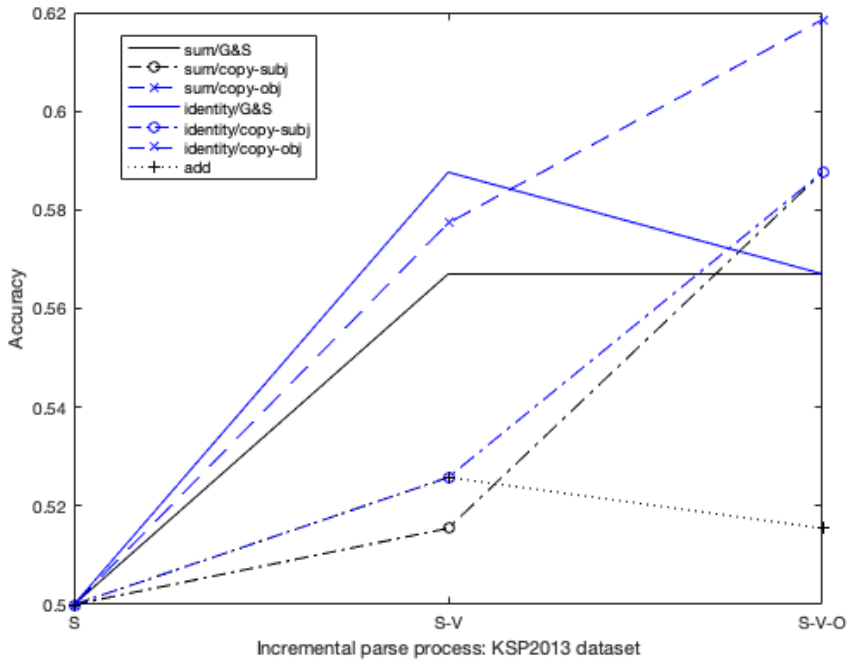


Figure 2: Mean verb disambiguation accuracy, as incremental parsing proceeds left-to-right through "S V O" sentences in the dataset of (Kartsaklis et al., 2013). The dotted line is the baseline standard addition method; the others show variants of our approach. Note that the *sum/copy-obj* and *identity/copy-obj* methods give identical average accuracy on this dataset, and thus share a line on the graph.

This work is described in full in (Purver et al., to appear), attached here as Appendix F.

5.2 Long-term context: topic and diachronic shift

Longer-term notions of context beyond the immediate sentence are required if we are to model medium-term effects (such as the effects of topic and salience on how words are interpreted within a particular document or conversation) and very long-term effects (such as the diachronic shift in word meaning as usage changes over the years).

Topic and salience effects We are currently approaching the problem of topic and salience in two ways. First, within this work package, we are analysing the ability of salience-based models to predict the contextual effects measured by our new CoSimLex dataset and evaluation task. Models such as that of McGregor et al. (2015) and Schockaert & Lee (2015) provide ways to identify subspaces within general embeddings spaces, and these can be used to emphasise particular salient parts of the meaning space, without the complexity of a BERT-like model. Our initial experiments using standard word2vec

embeddings in this way shows that positive correlations are achieved in our GWSC Subtask 1 evaluation (prediction of change direction); but currently this method fails to give good performance on Subtask 2 (prediction of absolute similarity ratings). We are continuing to investigate different variants.

Second, within WP4 and WP5, we are exploring the combination of topic models with BERT-style language models in order to better account for the effect of topical context on predictions; this will be described in the forthcoming Deliverables D3.4 (Final cross-lingual context and opinion analysis technology) and D4.5 (Final real-time multilingual news linking technology).

Diachronic effects An even longer-term notion of topic is important in *diachronic* analysis: modelling the changes in meaning and use of words over periods of years or centuries. Deliverable D1.3 described some of our initial work in this direction, in which we used mappings between time-specific embeddings spaces to model such changes. However, as described there, the availability of ELMo/BERT-style models changed our approach; we have since developed new ways of using those models to model diachronic effects, and work now performed in WP4 applies these using the models developed in Section 2; this will be described in the forthcoming Deliverable D4.7 (Final cross-lingual news viewpoints identification technology).

6 Conclusions and further work

This report summarizes the the work on contextual and dynamic embeddings performed in T1.2 of the EMBEDDIA project, with a focus on the second year.

As pointed out in the previous deliverable D1.3 (*Initial context-dependent and dynamic embeddings*), new developments since the EMBEDDIA project was proposed have produced effective models which are both context-dependent and dynamic. Our work here is therefore concentrated on the two most popular models that are able to produce contextual embeddings, ELMo and BERT. We developed new ELMo embeddings for our project languages, as presented in deliverable D1.3; here, we evaluate them against previously existing ELMoForManyLangs embeddings on two tasks, NER and dependency parsing, and show that they improve performance on both. We then present our newly trained trilingual BERT models and evaluate them on the same two tasks: our models perform significantly better than existing multilingual models, trained on 100 or more languages, and on par with or better than existing monolingual models. In future work we plan to explore training BERT models with different combinations of languages as well as our own monolingual models.

In deliverable D1.3 we introduced our goal to produce resources to allow for the intrinsic evaluation of contextual embeddings. Both the shared competition (SemEval2020 Task3: Graded Word Similarity in Context) and the dataset that enabled it (CoSimLex) were successfully finished during the second year. Here, we describe the new context-dependent approach it introduces, the metrics used for their evaluation, the annotation process and the CoSimLex dataset itself in English, Croatian, Slovene and Finnish. We describe the results of the public SemEval task, including the performance relative to baselines we created and the variety of submissions we received. We then apply this approach to evaluating our own EMBEDDIA ELMo and BERT models, showing that they outperform existing multilingual ELMo and BERT models, and that they would have performed competitively in the task itself.

Work in T1.2 will continue to build BERT monolingual embeddings for selected EMBEDDIA languages, and evaluating our models. More significantly, its outputs are now being taken forward in other EMBEDDIA work packages to enable cross-lingual and multilingual work and improved results, particularly in comment analysis (WP3), news article analysis (WP4) and software tools for the media industry (WP6). That work will be reported in later deliverables from the corresponding work packages.

7 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)
CroSloEng BERT embeddings	huggingface.co/EMBEDDIA/crosloengual-bert	Public(CC-BY 4.0)
FinEstEng BERT embeddings	huggingface.co/EMBEDDIA/finest-bert	Public(CC-BY 4.0)
LatLitEng BERT embeddings	huggingface.co/EMBEDDIA/litlat-bert	Public(CC-BY 4.0)
Crosslingual NER	github.com/EMBEDDIA/crosslingual-NER	Public (GPL v3)
SuPAR ELMo dependency parser	github.com/EMBEDDIA/supar-elmo	Public (GPL v3)
CoSimLex dataset	http://hdl.handle.net/11356/1308	Public (CC-BY-SA)
Slovenian SimLex	http://hdl.handle.net/11356/1309	Public (CC-BY-SA)

BERT models are also available as individual downloads from CLARIN: please see also the project website Outputs page at embeddia.eu/outputs which gives links to all pre-trained models and datasets.

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. and M. Robnik-Šikonja (2020). "High Quality ELMo Embeddings for Seven Less-Resourced Languages". Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020).	Published	Appendix A
Ulčar, M. and M. Robnik-Šikonja (2020). "FinEst BERT and CroSloEngual BERT: less is more in multilingual models." Proceedings of Text, Speech, and Dialogue (TSD 2020).	Published	Appendix B
Lau, J.H., C.S. Armendariz, S. Lappin, M. Purver, and C. Shu (2020). "How Furiously Can Colorless Green Ideas Sleep? Sentence Acceptability in Context". Transactions of the Association for Computational Linguistics 8: 296–310.	Published	Appendix C
Armendariz, C.S., M. Purver, M. Ulčar, S. Pollak, N. Ljubešić, M. Robnik-Šikonja, M. Granroth-Wilding, and K. Vaik (2020). "CoSimLex: A Resource for Evaluating Graded Word Similarity in Context". Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020).	Published	Appendix D
Armendariz, C.S., M. Ulčar, S. Pollak, N. Ljubešić, M. Robnik Šikonja, M. Granroth-Wilding, M. T. Pilehvar, I. Vulić, and M. Purver (2020). "SemEval 2020 Task 3: Graded Word Similarity in Context". In: Proceedings of the International Workshop on Semantic Evaluation (SemEval 2020).	Published	Appendix E
Purver, M., M. Sadrzadeh, R. Kempson, G. Wijnholds, and J. Hough (2021). Incremental composition in distributional semantics. Journal of Logic, Language and Information (to appear).	Accepted	Appendix F

References

Armendariz, C. S., Purver, M., Pollak, S., Ljubešić, N., Ulčar, M., Vulić, I., & Pilehvar, M. T. (2020, December). SemEval-2020 task 3: Graded word similarity in context. In *Proceedings of the fourteenth workshop on semantic evaluation* (pp. 36–49). Barcelona (online): International Committee for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/2020.semeval-1.3

Armendariz, C. S., Purver, M., Ulčar, M., Pollak, S., Ljubešić, N., & Granroth-Wilding, M. (2020, May). CoSimLex: A resource for evaluating graded word similarity in context. In *Proceedings of the 12th language resources and evaluation conference (LREC)* (pp. 5878–5886). Marseille, France: European Language Resources Association. Retrieved from https://www.aclweb.org/anthology/2020.lrec-1.720

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.

Bruni, E., Tran, N.-K., & Baroni, M. (2014). Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, *49*, 1–47.

Camacho-Collados, J., Pilehvar, M. T., Collier, N., & Navigli, R. (2017). SemEval-2017 Task 2: Multilingual and cross-lingual semantic word similarity. In *Proceedings of the 11th international workshop on semantic evaluation (semeval-2017)* (pp. 15–26).

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*.

Dargis, R., Auziņa, I., Bojārs, U., Paikens, P., & Znotiņš, A. (2018, may). Annotation of the corpus of the Saeima with multilingual standards. In D. Fišer, M. Eskevich, & F. de Jong (Eds.), *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018).* Paris, France: European Language Resources Association (ELRA).

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

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

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.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on world wide web* (p. 406–414). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/371920.372094 doi: 10.1145/371920.372094

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

Hill, F., Reichart, R., & Korhonen, A. (2015, December). SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, *41*(4), 665–695. Retrieved from http://www.aclweb.org/anthology/J15-4004 doi: 10.1162/COLI_a_00237

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).

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

Kartsaklis, D., Sadrzadeh, M., & Pulman, S. (2013, August). Separating disambiguation from composition in distributional semantics. In *Proceedings of the seventeenth conference on computational natural language learning (conll)* (pp. 114–123). Sofia, Bulgaria.

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

Kondratyuk, D., & Straka, M. (2019). 75 languages, 1 model: Parsing universal dependencies universally. In *Proceedings of the 2019 emnlp-ijcnlp* (pp. 2779–2795).

Lau, J. H., Armendariz, C. S., Lappin, S., Purver, M., & Shu, C. (2020). How furiously can colourless green ideas sleep? sentence acceptability in context. *Transactions of the Association for Computational Linguistics*, *8*, 296-310. Retrieved from https://doi.org/10.1162/tacl_a_00315 (Available as arXiV:2004.00881) doi: 10.1162/tacl_a_00315

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

McGregor, S., Agres, K., Purver, M., & Wiggins, G. (2015, December). From distributional semantics to conceptual spaces: A novel computational method for concept creation. *Journal of Artificial General Intelligence*, *6*(1), 55–86. doi: 10.1515/jagi-2015-0004

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

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, *38*(11), 39–41.

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

Nivre, J., Abrams, M., Agić, Ž., Ahrenberg, L., Antonsen, L., Aplonova, K., ... Zhu, H. (2018). *Universal dependencies 2.3.* Retrieved from http://hdl.handle.net/11234/1-2895 (LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University)

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., Sadrzadeh, M., Kempson, R., Wijnholds, G., & Hough, J. (to appear). Incremental composition in distributional semantics. *Journal of Logic, Language and Information*. Retrieved from http://www.eecs.qmul.ac.uk/~mpurver/papers/purver-et-al20jolli.pdf

Ruder, S., Peters, M. E., Swayamdipta, S., & Wolf, T. (2019, June). Transfer learning in natural language processing. In *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Tutorials* (pp. 15–18). Minneapolis, Minnesota: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/N19-5004 doi: 10.18653/v1/N19-5004

Schockaert, S., & Lee, J. H. (2015). Qualitative reasoning about directions in semantic spaces. In *Proceedings of the 24th international joint conference on artificial intelligence (ijcai)* (p. 3207-3213).

Tanvir, H., Kittask, C., & Sirts, K. (2020). *Estbert: A pretrained language-specific bert for estonian.*

Ulčar, M., & Robnik-Šikonja, M. (2020, May). High quality ELMo embeddings for seven less-resourced languages. In *Proceedings of the 12th language resources and evaluation conference (LREC)* (pp. 4731–4738). Marseille, France: European Language Resources Association. Retrieved from https://www.aclweb.org/anthology/2020.lrec-1.582

Ulčar, M., & Robnik-Šikonja, M. (2020). FinEst BERT and CroSloEngual BERT: less is more in multilingual models. In P. Sojka, I. Kopeček, K. Pala, & A. Horák (Eds.), *Text, speech, and dialogue TSD 2020* (Vol. 12284). Springer. Retrieved from https://doi.org/10.1007/978-3-030-58323-1_11 (Preprint available as https://arxiv.org/abs/2006.07890)

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.

Virtanen, A., Kanerva, J., Ilo, R., Luoma, J., Luotolahti, J., Salakoski, T., ... Pyysalo, S. (2019). Multilingual is not enough: BERT for Finnish. *arXiv preprint arXiv:1912.07076*.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... Rush, A. M. (2020, October). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: System demonstrations* (pp. 38–45). Online: Association for Computational Linguistics. Retrieved from https://www.aclweb.org/anthology/2020.emnlp-demos.6

Znotiņš, A., & Barzdiņš, G. (2020). Lvbert: Transformer-based model for latvian language understanding. In *Human language technologies–the baltic perspective: Proceedings of the ninth international conference baltic hlt 2020* (Vol. 328, p. 111).

Appendix A: High Quality ELMo Embeddings for Seven Less-Resourced Languages

Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 4731–4738
Marseille, 11–16 May 2020

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 tasks. 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 the 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., 2019). 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. While recently much more complex models such as BERT (Devlin et al., 2019) have further improved the results, ELMo is still useful for several reasons: its neural network only contains three layers and the explicit embedding vectors are therefore much easier to extract, it is faster to train and adapt to specific tasks.

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 the obtained results in Section 5. We present conclusion in Section 6 where we also outline plans for further work.

2. ELMo

Standard 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

[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/`.

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 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. The second LSTM 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 employed. 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 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 Latvian corpus (wikidump + common crawl), 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[2].

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 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 includes hrWaC 2.1 corpus[3] (Ljubešić and Klubička, 2014), Riznica[4] (Ćavar and Brozović Rončević, 2012), and articles of Croatian branch of Styria media house, made available to us through partnership in a joint project[5]. 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[6] (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[7], Finnish part of the CoNLL 2017 dataset, and Ylilauta downloadable version[8] (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, which is composed of Latvian Wikipedia and general webcrawl of Latvian webpages.

Lithuanian dataset is composed of Lithuanian Wikipedia articles from 2018, Lithuanian part of the DGT-UD corpus[9], and LtTenTen[10]. 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) of standard Slovene. 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.

[2] http://hdl.handle.net/11356/1277

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

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

[5] http://embeddia.eu

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

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

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

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

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

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

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

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[11] and its tokenizers 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[12] 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 the default values used to train the original English ELMo (large) model.

ELMo models were trained on machines with either two or three Nvidia GeForce GTX 1080 Ti GPUs. The training took roughly three weeks for each model. The exact time depended on the number of GPUs, size of the corpus, and other tasks running concurrently on the same machine.

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 in the category "capitals and countries", 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, in the category "comparative adjective", given the word pair "long : longer", we have an adjective in its base form and the same adjective in a comparative form. That task is 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 search for nearest neighbours using vector arithmetic, i.e. we compute the distance between vectors: d(vec(Finland), vec(Helsinki)) and search for the 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 by Ulčar and Robnik-Šikonja (2019), which are based on the dataset by Mikolov et al. (2013a) and are available at Clarin repository (Ulčar et al., 2019).

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 tackled this issue with two different approaches.

5.1.1. Average over Word Embeddings

In the first approach, we calculated ELMo embeddings for each token of a large corpus and then averaged the vectors of all the occurences of each word, effectively creating non-contextual word embeddings. For each language, we used language specific Wikipedia as the corpus. The positive side of this approach is that it accounts for many different occurences of each word in various contexts and thus provides sensible embeddings. The downsides are that by averaging we lose context information, and that the process is lengthy, taking several days per language. We performed this approach on three languages: Croatian, Slovenian and English. We used these non-contextual ELMo embeddings in the word analogy task in the same way as any other non-contextual embeddings.

We used the nearest neighbor metric to find the closest candidate word. If we find the correct word among the n closest words, we consider that entry as successfully identified. The proportion of correctly identified words forms a measure called accuracy@n, which we report as the result.

In Table 2, we show the results for different layers of ELMo

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

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

models used as embeddings and their comparison with the baseline fastText embeddings. Among ELMo embeddings, the best result on syntactic categories are obtained by using the vectors after 2nd layer (LSTM1), while the best result on semantic categories are obtained using vectors after the 3rd layer of the neural model (LSTM2). Compared to fastText, the results vary from language to language. In English, fastText embeddings outperform ELMo in both semantic and syntactic categories. In Slovenian, ELMo embeddings outperform fastText embeddings, significantly so in syntactic categories. In Croatian, ELMo outperforms fastText on syntactic categories, but on semantic categories fastText is a bit better.

Layer	category	Croatian	Slovenian	English
CNN	semantic	0.081	0.059	0.120
	syntactic	0.475	0.470	0.454
LSTM1	semantic	0.219	0.305	0.376
	syntactic	0.663	0.677	0.595
LSTM2	semantic	0.214	0.306	0.404
	syntactic	0.604	0.608	0.545
fastText	semantic	0.284	0.239	0.667
	syntactic	0.486	0.437	0.626

Table 2: The embeddings quality measured on the word analogy task, using accuracy@1 score, where 200,000 most common words were considered. The embeddings for each word were obtained by averaging the embeddings of each occurence in the Wikipedia. Results are shown for each layer of ELMo model 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 then these results are averaged).

5.1.2. Analogy in a Simple Sentence

In the second approach to analogy evaluation, we 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).

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.

As Figure 1 shows, the Latvian ELMo model from ELMoForManyLangs project performs significantly worse than our ELMo Latvian model (named EMBEDDIA) on all categories of word analogy task. 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 forms 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 3. The embeddings after the first LSTM layer (LSTM1) 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 they still outperform CNN layer embeddings in semantic categories. Latvian ELMo performs worse compared to other languages we trained, especially in semantic categories, presumably due to the 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. The English results strongly differ from the results of the first method (Table 2). The simple sentence used might have caused more problems in English than in other languages, but additional evaluation in various contexts and other evaluation tasks would be needed to better explain these results.

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 (NE) 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 la-

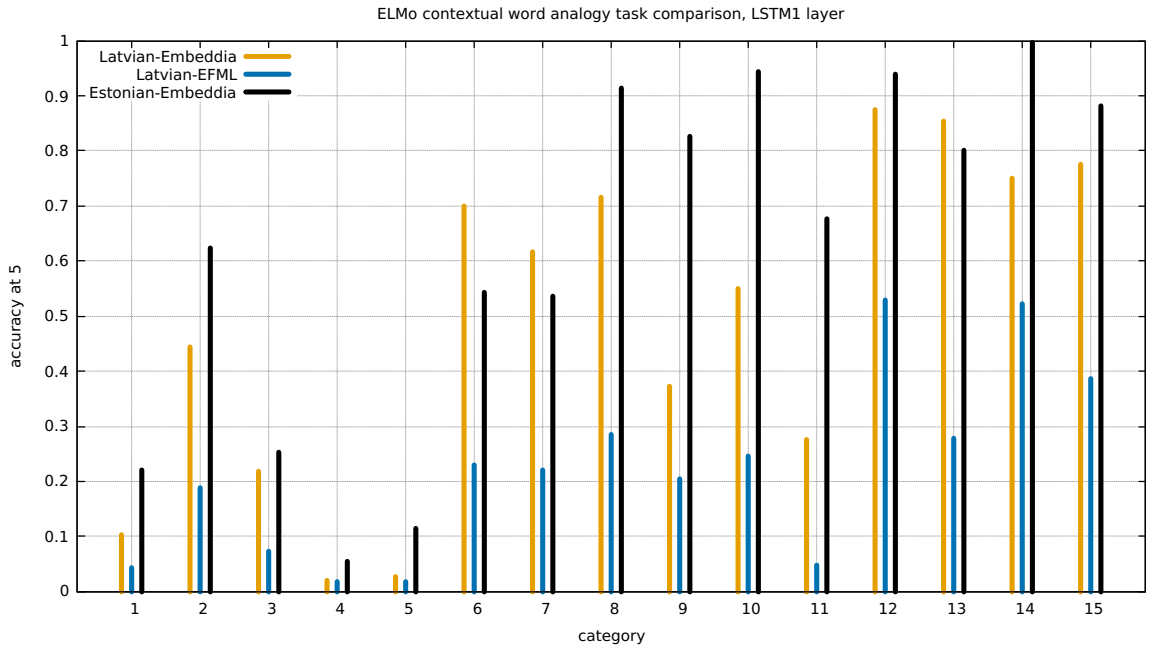


Figure 1: Comparison of Latvian ELMo model by ELMoForManyLangs (blue, Latvian-EFML), Latvian ELMo model trained by us (yellow, Latvian-Embeddia), and Estonian ELMo model trained by us (black, Estonian-Embeddia). The performance is measured as accuracy@5 on word analogy task, where categories 1 to 5 are semantic, and categories 6 to 15 are syntactic. The embeddings use weights of the first biLM layer LSTM1 (i.e. the second layer overall).

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

Table 3: The embeddings quality measured on the word analogy task, using accuracy@5 score. Each language is represented with its 2-letter ISO code (first column). 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 results are then averaged).

Language	PER	LOC	ORG	density	N
Croatian	10241	7445	11216	0.057	506457
Estonian	8490	6326	6149	0.096	217272
Finnish	3402	2173	11258	0.087	193742
Latvian	5615	2643	3341	0.085	137040
Lithuanian	2101	2757	2126	0.076	91983
Slovenian	4478	2460	2667	0.049	194667
Swedish	3976	1797	1519	0.047	155332
English	17050	12316	14613	0.146	301418

Table 4: The number of tokens labeled with each label (PER, LOC, ORG), the density of these labels (their sum divided by the number of all tokens) and the number of all tokens (N) for datasets in all languages.

beled 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 4.

To measure the performance of ELMo embeddings on the NER task we proceeded as follows. We split the NER datasets into training (90% of sentences) and testing (10% of sentences) set. We embedded text sentence by sentence, producing three vectors (one from each ELMo layer) for each token in a sentence. For prediction of NEs, we trained a neural network model, where we used three input layers (one embedding vector for each input). We then averaged the input layers, such that the model learned the averaging weights during the training. Next, we added two BiLSTM layers with 2048 LSTM cells each, followed by a time distributed softmax layer with 4 neurons.

We used ADAM optimiser (Kingma and Ba, 2014) with the learning rate 10^{-4} and 10^{-5} learning rate decay. We used categorical cross-entropy as a loss function and trained each model for 10 epochs (except Slovenian with EFML embeddings, where we trained for 5 epochs, since it gives

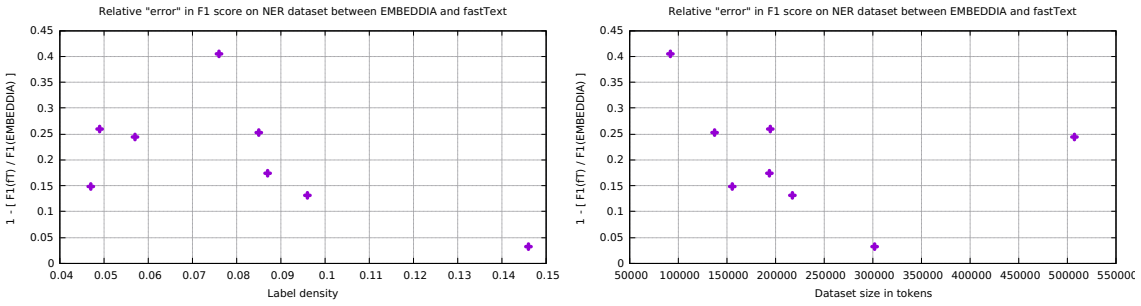


Figure 2: Comparison between fastText and EMBEDDIA ELMo embeddings on NER task. We show the relative difference (error) between the F_1 scores, in relation to the label density (left) and dataset size (right).

a much better score ($0.82F_1$ vs. $0.68F_1$)). 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) in Table 5.

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 NEs using them. The architecture of the model using fastText embeddings is the same as the one using ELMo embeddings, except that we have one input layer, which receives 300 dimensional fastText embedding vectors. We also compared performance with ELMoForManyLangs (EFML) embeddings, using the same architecture as with our ELMo embeddings. In all cases (ELMo, EFML and fastText), we trained and evaluated prediction models five times and averaged the results due to randomness in initialization of neural network models. There is no Lithuanian EFML model, so we could not compare the two ELMo models on that language.

Both ELMo embeddings (EFML and our EMBEDDIA) show significant improvement in performance on NER task over fastText embeddings on all languages, except English (Table 5). In English, there is still improvement, but a smaller one, in part due to already high performance using fastText embeddings.

The difference between our ELMo embeddings and EFML embeddings is smaller on the NER task than on the word analogy task. On Latvian dataset, the performance is equal, while we have observed a significant difference on the word analogy task (Figure 1). Our ELMo embedding models, however, show larger improvement over EFML on NER tasks in some other languages, like Croatian.

We compared the difference in performance of EMBEDDIA ELMo embeddings and fastText embeddings as a function of dataset size and label density (Figure 2). Barring one outlier, there is a slight negative correlation with regard to the dataset size, but no correlation with label density. We compared the EFML and EMBEDDIA ELMo embeddings in the same manner (Figure 3), with no apparent correlation.

[13] https://fasttext.cc/

Language	fastText	EFML	EMBEDDIA
Croatian	0.62	0.73	0.82
Estonian	0.79	0.89	0.91
Finnish	0.76	0.88	0.92
Latvian	0.62	0.83	0.83
Lithuanian	0.44	N/A	0.74
Slovenian	0.63	0.82	0.85
Swedish	0.75	0.85	0.88
English	0.89	0.90	0.92

Table 5: The results of NER evaluation task. The scores are macro average F_1 scores of the three named entity classes, excluding score for class "Other". The columns show fastText, ELMoForManyLangs (EFML), and EMBEDDIA ELMo embeddings.

6. Conclusion

We prepared high quality 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 can be improved for some downstream tasks. 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 comparison with the existing ELMoForManyLangs embeddings, our new EMBEDDIA ELMo embeddings show a big improvement on the analogy task, and a significant improvement on the NER task.

For a more thorough analysis of our ELMo embeddings, more downstream tasks shall be considered. Unfortunately, no such task currently exist for the majority of the seven processed languages.

As future work, we will use the produced contextual embeddings on the problems of news media industry. We plan

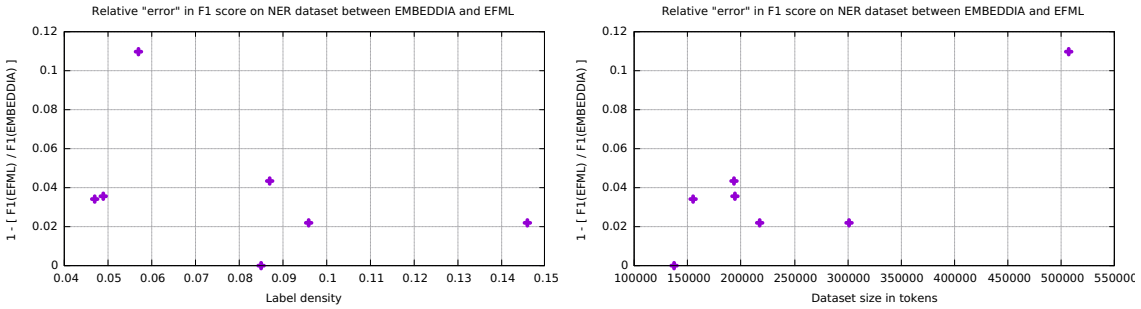


Figure 3: Comparison between EFML and EMBEDDIA ELMo embeddings on NER task. We show the relative difference (error) between the F_1 scores, in relation to the label density (left) and dataset size (right).

to build and evaluate more complex models, such as BERT (Devlin et al., 2019). The pretrained EMBEDDIA ELMo models are publicly available on the CLARIN repository[14].

7. Acknowledgments

The work was partially supported by the Slovenian Research Agency (ARRS) through core research programme P6-0411 and research project J6-8256 (New grammar of contemporary standard Slovene: sources and methods). 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.

8. 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. (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.

[14] http://hdl.handle.net/11356/1277

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.

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)*.

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*.

Ulčar, M. and Robnik-Šikonja, M. (2019). Multilingual Culture-Independent Word Analogy Datasets. *arXiv preprint 1911.10038*.

9. Language Resource References

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.

Krek, S., Erjavec, T., Repar, A., Čibej, J., Arhar, S., Gantar, P., Kosem, I., Robnik-Šikonja, 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. https://viri.cjvt.si/gigafida.

Ulčar, M., Vaik, K., Lindström, J., Linde, D., Dailidėnaitė, M., and Šumakov, A. (2019). Multilingual Culture-Independent Word Analogy Datasets. Slovenian language resource repository CLARIN.SI http://hdl.handle.net/11356/1261.

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

Appendix B: FinEst BERT and CroSloEngual BERT: less is more in multilingual models

FinEst BERT and CroSloEngual BERT: less is more in multilingual models

Matej Ulčar and Marko Robnik-Šikonja

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

Abstract. Large pretrained masked language models have become state-of-the-art solutions for many NLP problems. The research has been mostly focused on English language, though. While massively multilingual models exist, studies have shown that monolingual models produce much better results. We train two trilingual BERT-like models, one for Finnish, Estonian, and English, the other for Croatian, Slovenian, and English. We evaluate their performance on several downstream tasks, NER, POS-tagging, and dependency parsing, using the multilingual BERT and XLM-R as baselines. The newly created FinEst BERT and CroSloEngual BERT improve the results on all tasks in most monolingual and cross-lingual situations.

Keywords: contextual embeddings, BERT model, less-resourced languages, NLP

1 Introduction

In natural language processing (NLP), a lot of research focuses on numeric word representations. Static pretrained word embeddings like word2vec [11] are recently replaced by dynamic, contextual embeddings, such as ELMo [13] and BERT [3]. These generate a word vector based on the context the word appears in, mostly using the sentence as the context.

Large pretrained masked language models like BERT [3] and its derivatives achieve state-of-the-art performance when fine-tuned for specific NLP tasks. The research into these models has been mostly limited to English and a few other well-resourced languages, such as Chinese Mandarin, French, German, and Spanish. However, two massively multilingual masked language models have been released: a multilingual BERT (mBERT) [3], trained on 104 languages, and newer even larger XLM-RoBERTa (XLM-R) [2], trained on 100 languages. While both, mBERT and XLM-R, achieve good results, it has been shown that monolingual models significantly outperform multilingual models [20, 10]. In our work, we reduced the number of languages in multilingual models to three, two similar less-resourced languages from the same language family, and English. The main reasons for this choice are to better represent each language, and keep sensible

sub-word vocabulary, as shown by Virtanen et al. [20]. We decided against production of monolingual models, because we are interested in using the models in multilingual sense and for cross-lingual knowledge transfer. By including English in each of the two models, we expect to better transfer existing prediction models from English to involved less-resourced languages. Additional reason against purely monolingual models for less-resourced languages is the size of training corpora, i.e. BERT-like models use transformer architecture which is known to be data hungry.

We thus trained two multilingual BERT models: FinEst BERT was trained on Finnish, Estonian, and English, while CroSloEngual BERT was trained on Croatian, Slovenian, and English. In the paper, we present the creation and evaluation of these models, which required considerable computational resources, unavailable to most NLP researchers. We make the models which are valuable resources for the involved less-resourced languages publicly available[1].

2 Training data and preprocessing

BERT models require large quantities of monolingual data. In Section 2.1 we first describe the corpora used, followed by a short description of their preprocessing in Section 2.2.

2.1 Datasets

We trained two new BERT models from five languages: Finnish, Estonian, Slovenian, Croatian and English. To obtain high-quality models, we used large monolingual corpora for each language, some of them unavailable to the general public. For English, large corpora are readily available and they are much larger than for other languages. However, high-quality English language models already exist and English is not the main focus of this research, we therefore did not use all available English corpora in order to prevent English from overwhelming the other languages in our models. Some corpora are available online under permissive licences, others are available only for research purposes or have limited availability. The corpora used in training are a mix of news articles and general web crawl, which we preprocessed and deduplicated. Details about the training set sizes are presented in Table 1, while their description can be found in works on the involved less-resourced languages, e,g., [18].

2.2 Preprocessing

Before using the corpora, we deduplicated them for each language separately, using the Onion (ONe Instance ONly) tool[2]. We applied the tool on sentence

[1] CroSloEngual BERT: `http://hdl.handle.net/11356/1317`
FinEst BERT: `http://urn.fi/urn:nbn:fi:lb-2020061201`

[2] `http://corpus.tools/wiki/Onion`

Table 1. The training corpora sizes in number of tokens and the ratios for each language.

Model	CroSloEngual	FinEst
Croatian	31%	0%
Slovenian	23%	0%
English	47%	63%
Estonian	0%	13%
Finnish	0%	25%
Tokens	$5.9 \cdot 10^9$	$3.7 \cdot 10^9$

Table 2. The sizes of corpora subsets in millions of tokens used to create wordpiece vocabularies.

Language	FinEst	CroSloEngual
Croatian	/	27
Slovenian	/	28
English	157	23
Estonian	75	/
Finnish	97	/

level for those corpora that did have sentences shuffled, and on paragraph level for the rest. As parameters, we used 9-grams with duplicate content threshold of 0.9.

BERT models are trained on subword (wordpiece) tokens. We created a wordpiece vocabulary using bert-vocab-builder tool[3], which is built upon tensor2tensor library [19]. We did not process the whole corpora in creating the wordpiece vocabulary, but only a smaller subset. To balance the language representation in vocabulary, we used samples from each language. The sizes of corpora subsets are shown in Table 2. The created wordpiece vocabularies contain 74,986 tokens for FinEst and 49,601 tokens for CroSloEngual model.

3 Architecture and training

We trained two BERT multilingual models. FinEst BERT was trained on Finnish, Estonian, and English corpora, with altogether 3.7 billion tokens. CroSloEngual BERT was trained on Croatian, Slovenian, and English corpora with together 5.9 billion tokens.

Both models use bert-base architecture [3], which is a 12-layer bidirectional transformer encoder with the hidden layer size of 768 and altogether 110 million parameters. We used the whole word masking for the masked language model training task. Both models are cased, i.e. the case information was preserved. We followed the hyper-parameters settings of Devlin et al. [3], except for the batch size and total number of steps. We trained the models for approximately 40 epochs with maximum sequence length of 128 tokens, followed by approximately 4 epochs with maximum sequence length of 512 tokens. The exact number of steps was calculated using the expression:

$$s = \frac{N_{tok} \cdot E}{b \cdot \lambda}$$

, where s is the number of steps the models were trained for, N_{tok} is the number of tokens in the train corpora, E is the desired number of epochs (in our case 40 and 4), b is the batch size, and λ is the maximum sequence length.

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

We trained FinEst BERT on a single Google Cloud TPU v3 for a total of 1.24 million steps where the first 1.13 million steps used the batch size of 1024 and sequence length 128, and the last 113 thousand steps used the batch size 256 and sequence length 512. Similarly, CroSloEngual BERT was trained on a single Google Cloud TPU v2 for a total of 3.96 million steps, where the first 3.6 million steps used the batch size of 512 and sequence length 128, and the last 360 thousand steps were trained with the batch size 128 and sequence length 512. Training took approximately 2 weeks for FinEst BERT and approximately 3 weeks for CroSloEngual BERT.

4 Evaluation

We evaluated the two new BERT models on three downstream evaluation tasks available for the four involved less-resourced languages: named entity recognition (NER), part-of-speech tagging (POS), and dependency parsing (DP). We compared both models with BERT-base-multilingual-cased model (mBERT) on sensible languages, i.e. FinEst BERT was compared with mBERT on Finnish, Estonian, and English, while CroSloEngual BERT was compared with mBERT on Croatian, Slovenian, and English.

4.1 Named Entity Recognition

Named entity recognition (NER) task is a sequence labeling task, which tries to correctly identify and classify each token from an unstructured text into one of the predefined named entity (NE) classes or, if the token is not part of a NE, to classify it as not a named entity. Most common named entity classes are personal names, locations and organizations. We used various datasets, which do not cover the same set of classes. We therefore adapted the datasets to allow a more direct comparison between languages, by reducing them to the four labels they all have in common: PER (person), LOC (location), ORG (organization), and O (other). All tokens, which are not named entities or belong to any NE class other than person, location or organization, were labeled as 'O'.

For Croatian and Slovenian, we used data from hr500k [9] and ssj500k [7], respectively. Not all sentences in ssj500k are annotated, so we excluded those that are not annotated. English dataset comes from CoNLL 2013 shared task [17]. For Finnish we used Finnish News Corpus for NER [15], and for Estonian dataset we used Nimeüksuste korpus [8]. The statistics of each dataset are shown in Table 3.

To evaluate the performance of BERT embeddings on the NER task we trained NER models using Huggingface's Transformer library, basing the code on their NER example[4]. We fine-tuned each of our BERT models with an added token classification head for 3 epochs on the NER data. We compared the results with BERT-base-multilingual-cased (mBERT) model, which we fine-tuned with exactly the same parameters on the same data.

[4] `https://github.com/huggingface/transformers/tree/master/examples/ner`

Language	PER	LOC	ORG	Density	N
Croatian	10241	7445	11216	0.057	506457
English	17050	12316	14613	0.146	301418
Estonian	8490	6326	6149	0.096	217272
Finnish	3402	2173	11258	0.087	193742
Slovenian	4478	2460	2667	0.049	194667

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

Train lang	Test lang	mBERT	CroSloEngual
Croatian	Croatian	0.795	0.894
Slovenian	Slovenian	0.903	0.917
English	English	0.940	0.949
Croatian	English	0.793	0.866
English	Croatian	0.638	0.798
Slovenian	English	0.781	0.833
English	Slovenian	0.736	0.843
Croatian	Slovenian	0.825	0.908
Slovenian	Croatian	0.755	0.847

Table 4. The results of NER evaluation task on Croatian, Slovenian, and English. The scores are average F_1 scores of the three named entity classes. A NER model was trained on "train language" dataset and tested on "test language" dataset using two different BERT models for all possible combinations of train and test languages.

We evaluated the models in a monolingual setting (training and testing on the same language) and a crosslingual setting (training on one language, testing on another). We present the results as macro average F_1 scores of the three NE classes, excluding 'O' label. Comparison between CroSloEngual BERT and mBERT is shown in Table 4, comparison between FinEst BERT and mBERT is shown in Table 5.

The difference in performance of each BERT on English data is negligible. In other languages, our models outperform the multilingual BERT, the difference is especially large in Croatian. In crosslingual setting, both FinEst BERT and CroSloEngual BERT show a significant improvement over mBERT, especially when one of the two languages is English. This leads us to believe that multilingual BERT models with fewer languages are more suitable for crosslingual knowledge transfer.

Train lang	Test lang	mBERT	FinEst
Finnish	Finnish	0.922	0.959
Estonian	Estonian	0.906	0.930
English	English	0.940	0.942
Finnish	English	0.692	0.810
English	Finnish	0.770	0.901
Estonian	English	0.765	0.815
English	Estonian	0.762	0.839
Finnish	Estonian	0.795	0.879
Estonian	Finnish	0.839	0.912

Table 5. The results of NER evaluation task on Finnish, Estonian, and English. The scores are average F_1 scores of the three named entity classes. A NER model was trained on "train language" dataset and tested on "test language" dataset using two different BERT models for all possible combinations of train and test languages.

4.2 Part-of-speech tagging and dependency parsing

We evaluated BERT models on two more classification tasks: part-of-speech (POS) tagging and dependency parsing. In the POS tagging task we attempt to correctly classify each token within a given set of grammatical categories (verb, adjective, punctuation, adverb, noun, etc.) Dependency parsing task attempts to predict the tree structure, representing the syntactic relations between words in a given sentence.

We trained classifiers on universal dependencies (UD) treebank datasets, using universal part-of-speech (UPOS) tag set. For Croatian, we used treebank by Agić and Ljubešić [1]. For English, we used A Gold Standard Dependency Corpus [16]. For Estonian, we used Estonian Dependency Treebank [12], converted to UD. Finnish treebank used is based on the Turku Dependency Treebank [5], which was also converted to UD [14]. Slovenian treebank [4] is based on the ssj500k corpus [7].

We used Udify tool [6] to train both POS tagger and dependency parsing classifiers at the same time. We finetuned each BERT model for 80 epochs on the treebank data. We kept the tool parameters at default values, except for "warmup_steps" and "start_step" values, which we changed to equal the number of training batches in one epoch.

We present the results of POS tagging as UPOS accuracy score in Table 6 and Table 7. The difference in performance between BERT models is very small on this task. FinEst and CroSloEngual BERTs perform slightly better than mBERT on all languages in monolingual setting, except Croatian, where mBERT and CroSloEngual BERT are equal. The differences are more pronounced in cross-lingual setting. When training on Slovenian, Finnish or Estonian data and testing on English data CroSloEngual and FinEst BERT significantly outperform mBERT. On the other hand, when training on English and testing Croatian, mBERT outperforms CroSloEngual BERT.

Train lang.	Test lang.	mBERT	CroSloEngual
Croatian	Croatian	0.983	0.983
English	English	0.969	0.972
Slovenian	Slovenian	0.987	0.991
English	Croatian	0.876	0.869
English	Slovenian	0.857	0.859
Croatian	English	0.750	0.756
Croatian	Slovenian	0.917	0.934
Slovenian	English	0.686	0.723
Slovenian	Croatian	0.920	0.935

Table 6. The embeddings quality measured on the UPOS tagging task, using UPOS accuracy score for FinEst BERT, CroSloEngual BERT and BERT-base-multilingual-cased (mBERT).

Train lang.	Test lang.	mBERT	FinEst
English	English	0.969	0.970
Estonian	Estonian	0.972	0.978
Finnish	Finnish	0.970	0.981
English	Estonian	0.852	0.878
English	Finnish	0.847	0.872
Estonian	English	0.688	0.808
Estonian	Finnish	0.872	0.913
Finnish	English	0.535	0.701
Finnish	Estonian	0.888	0.919

Table 7. The embeddings quality measured on the UPOS tagging task, using UPOS accuracy score for FinEst BERT, CroSloEngual BERT and BERT-base-multilingual-cased (mBERT).

We present the results of dependency parsing task as unlabeled attachement score (UAS) and labeled attachment score (LAS). In monolingual setting CroSloEngual BERT shows improvement over mBERT on all three languages (Table 8) with the highest improvement on Slovenian and only a marginal improvement on English. FinEst BERT outperforms mBERT on Estonian and Finnish, with the biggest margin being on the Finnish data (Table 9). FinEst BERT and mBERT perform equally on English data.

In crosslingual setting, the results are similar to those seen on the POS tagging task. Major improvements of FinEst BERT and CroSloEngual BERT over mBERT in English-Estonian, English-Finnish and English-Slovenian pairs, minor improvements in Estonian-Finnish and Croatian-Slovenian pairs. Again, mBERT outperformed CroSloEngual BERT when dependency parser was trained on English data and tested on Croatian data.

Train language	Test language	mBERT UAS	mBERT LAS	CroSloEngual UAS	CroSloEngual LAS
Croatian	Croatian	0.930	0.891	0.940	0.903
English	English	0.917	0.894	0.922	0.899
Slovenian	Slovenian	0.938	0.922	0.957	0.947
English	Croatian	0.824	0.724	0.822	0.725
English	Slovenian	0.830	0.719	0.848	0.736
Croatian	English	0.759	0.627	0.782	0.657
Croatian	Slovenian	0.880	0.802	0.912	0.840
Slovenian	English	0.741	0.578	0.794	0.648
Slovenian	Croatian	0.861	0.773	0.891	0.810

Table 8. The embeddings quality measured on the dependency parsing task. Results are given as UAS and LAS for CroSloEngual BERT and BERT-base-multilingual-cased (mBERT).

Train language	Test language	mBERT UAS	mBERT LAS	FinEst UAS	FinEst LAS
English	English	0.917	0.894	0.918	0.895
Estonian	Estonian	0.880	0.848	0.909	0.882
Finnish	Finnish	0.898	0.867	0.933	0.915
English	Estonian	0.697	0.531	0.768	0.591
English	Finnish	0.706	0.561	0.781	0.624
Estonian	English	0.633	0.492	0.726	0.567
Estonian	Finnish	0.784	0.695	0.864	0.801
Finnish	English	0.543	0.433	0.684	0.558
Finnish	Estonian	0.782	0.691	0.852	0.778

Table 9. The embeddings quality measured on the dependency parsing task. Results are given as UAS and LAS for FinEst BERT and BERT-base-multilingual-cased (mBERT).

5 Conclusion

We built two large pretrained trilingual BERT-based masked language models, Croatian-Slovenian-English and Finnish-Estonian-English. We showed that the new CroSloEngual and FinEst BERTs perform substantially better than massively multilingual mBERT on the NER task in both monolingual and cross-lingual setting. The results on POS tagging and DP tasks show considerable improvement of the proposed models for several monolingual and cross-lingual pairs, while they are never worse than mBERT.

In future, we plan to investigate different combinations and proportions of less-resourced languages in creation of pretrained BERT-like models, and use the newly trained BERT models on the problems of news media industry.

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). Research was supported with Cloud TPUs from Google's TensorFlow Research Cloud (TFRC).

Bibliography

[1] Željko Agić and Nikola Ljubešić. Universal dependencies for Croatian (that work for Serbian, too). In *The 5th Workshop on Balto-Slavic Natural Language Processing*, pages 1–8, 2015.

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

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Kaja Dobrovoljc, Tomaž Erjavec, and Simon Krek. The universal dependencies treebank for Slovenian. In *Proceeding of the 6th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2017)*, 2017.

[5] K. Haverinen, J. Nyblom, T. Viljanen, V. Laippala, S. Kohonen, A. Missilä, S. Ojala, T. Salakoski, and F. Ginter. Building the essential resources for Finnish: the Turku dependency treebank. *LREC*, 2013.

[6] Dan Kondratyuk and Milan Straka. 75 languages, 1 model: Parsing universal dependencies universally. In *Proceedings of the 2019 EMNLP-IJCNLP*, pages 2779–2795, 2019.

[7] Simon Krek, Kaja Dobrovoljc, Tomaž Erjavec, Sara Može, Nina Ledinek, Nanika Holz, Katja Zupan, Polona Gantar, Taja Kuzman, Jaka Čibej, Špela Arhar Holdt, Teja Kavčič, Iza Škrjanec, Dafne Marko, Lucija Jezeršek, and Anja Zajc. Training corpus ssj500k 2.2, 2019. Slovenian language resource repository CLARIN.SI.

[8] Sven Laur. Nimeüksuste korpus. Center of Estonian Language Resources, 2013.

[9] Nikola Ljubešić, Filip Klubička, Željko Agić, and Ivo-Pavao Jazbec. New inflectional lexicons and training corpora for improved morphosyntactic annotation of Croatian and Serbian. In *Proceedings of the LREC 2016*, 2016.

[10] Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT: a tasty French language model. *arXiv preprint arXiv:1911.03894*, 2019.

[11] Tomas Mikolov, Quoc V Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint 1309.4168*, 2013.

[12] Kadri Muischnek, Kaili Müürisep, and Tiina Puolakainen. Estonian Dependency Treebank: from Constraint Grammar tagset to Universal Dependencies. In *Proceedings of LREC 2016*, 2016.

[13] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[14] Sampo Pyysalo, Jenna Kanerva, Anna Missilä, Veronika Laippala, and Filip Ginter. Universal dependencies for Finnish. In *Proceedings of NoDaLiDa 2015*, 2015.

[15] Teemu Ruokolainen, Pekka Kauppinen, Miikka Silfverberg, and Krister Lindén. A Finnish news corpus for named entity recognition. *Lang Resources & Evaluation*, 54(1):247–272, 2020.

[16] Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Christopher D. Manning. A gold standard dependency corpus for English. In *Proceedings of LREC-2014*, 2014.

[17] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada, 2003.

[18] Matej Ulčar and Marko Robnik-Šikonja. High quality elmo embeddings for seven less-resourced languages. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4731–4738, Marseille, France, May 2020. European Language Resources Association.

[19] Ashish Vaswani, Samy Bengio, Eugene Brevdo, Francois Chollet, Aidan Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, et al. Tensor2tensor for neural machine translation. In *Proceedings of the AMT*, pages 193–199, 2018.

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

Appendix C: How Furiously Can Colorless Green Ideas Sleep? Sentence Acceptability in Context

How Furiously Can Colorless Green Ideas Sleep? Sentence Acceptability in Context

Jey Han Lau[1,7] **Carlos Armendariz**[2] **Shalom Lappin**[2,3,4]
Matthew Purver[2,5] **Chang Shu**[6,7]

[1]The University of Melbourne [2]Queen Mary University of London
[3]University of Gothenburg [4]King's College London
[5]Jožef Stefan Institute [6]University of Nottingham Ningbo China [7]DeepBrain
jeyhan.lau@gmail.com, c.santosarmendariz@qmul.ac.uk
shalom.lappin@gu.se, m.purver@qmul.ac.uk, scxcs1@nottingham.edu.cn

Abstract

We study the influence of context on sentence acceptability. First we compare the acceptability ratings of sentences judged in isolation, with a relevant context, and with an irrelevant context. Our results show that context induces a cognitive load for humans, which compresses the distribution of ratings. Moreover, in relevant contexts we observe a discourse coherence effect that uniformly raises acceptability. Next, we test unidirectional and bidirectional language models in their ability to predict acceptability ratings. The bidirectional models show very promising results, with the best model achieving a new state-of-the-art for unsupervised acceptability prediction. The two sets of experiments provide insights into the cognitive aspects of sentence processing and central issues in the computational modeling of text and discourse.

1 Introduction

Sentence *acceptability* is the extent to which a sentence appears natural to native speakers of a language. Linguists have often used this property to motivate grammatical theories. Computational language processing has traditionally been more concerned with *likelihood*—the probability of a sentence being produced or encountered. The question of whether and how these properties are related is a fundamental one. Lau et al. (2017b) experiment with unsupervised language models to predict acceptability, and they obtained an encouraging correlation with human ratings. This raises foundational questions about the nature of linguistic knowledge: If probabilistic models can acquire knowledge of sentence acceptability from raw texts, we have prima facie support for an alternative view of language acquisition that does not rely on a categorical grammaticality component.

It is generally assumed that our perception of sentence acceptability is influenced by *context*. Sentences that may appear odd in isolation can become natural in some environments, and sentences that seem perfectly well formed in some contexts are odd in others. On the computational side, much recent progress in language modeling has been achieved through the ability to incorporate more document context, using broader and deeper models (e.g., Devlin et al., 2019; Yang et al., 2019). While most language modeling is restricted to individual sentences, models can benefit from using additional context (Khandelwal et al., 2018). However, despite the importance of context, few psycholinguistic or computational studies systematically investigate how context affects acceptability, or the ability of language models to predict human acceptability judgments.

Two recent studies that explore the impact of document context on acceptability judgments both identify a *compression* effect (Bernardy et al., 2018; Bizzoni and Lappin, 2019). Sentences perceived to be low in acceptability when judged without context receive a boost in acceptability when judged within context. Conversely, those with high out-of-context acceptability see a reduction in acceptability when context is presented. It is unclear what causes this compression effect. Is it a result of cognitive load, imposed by additional

Transactions of the Association for Computational Linguistics, vol. 8, pp. 296–310, 2020. https://doi.org/10.1162/tacl_a_00315
Action Editor: George Foster. Submission batch: 10/2019; Revision batch: 1/2020; Published 6/2020.

processing demands, or is it the consequence of an attempt to identify a discourse relation between context and sentence?

We address these questions in this paper. To understand the influence of context on human perceptions, we ran three crowdsourced experiments to collect acceptability ratings from human annotators. We develop a methodology to ensure comparable ratings for each *target sentence* in isolation (without any context), in a relevant three-sentence context, and in the context of sentences randomly sampled from another document. Our results replicate the compression effect, and careful analyses reveal that both cognitive load and discourse coherence are involved.

To understand the relationship between sentence acceptability and probability, we conduct experiments with unsupervised language models to predict acceptability. We explore traditional unidirectional (left-to-right) recurrent neural network models, and modern bidirectional transformer models (e.g., BERT). We found that bidirectional models consistently outperform unidirectional models by a wide margin, calling into question the suitability of left-to-right bias for sentence processing. Our best bidirectional model achieves simulated human performance on the prediction task, establishing a new state-of-the-art.

2 Acceptability in Context

2.1 Data Collection

To understand how humans interpret acceptability, we require a set of sentences with varying degrees of well-formedness. Following previous studies (Lau et al., 2017b; Bernardy et al., 2018), we use round-trip machine translation to introduce a wide range of infelicities into naturally occurring sentences.

We sample 50 English (target) sentences and their contexts (three preceding sentences) from the English Wikipedia.[1] We use Moses to translate the target sentences into four languages (Czech, Spanish, German, and French) and then back to English.[2] This produces 250 sentences in total (5 languages including English) for our *test* set. Note that we only do round-trip translation for the target sentences; the contexts are not modified.

We use Amazon Mechanical Turk (AMT) to collect acceptability ratings for the target sentences.[3] We run three experiments where we expose users to different types of context. For the experiments, we split the test set into 25 HITs of 10 sentences. Each HIT contains 2 original English sentences and 8 round-trip translated sentences, which are different from each other and not derived from either of the originals. Users are asked to rate the sentences for naturalness on a 4-point ordinal scale: bad (1.0), not very good (2.0), mostly good (3.0), and good (4.0). We recruit 20 annotators for each HIT.

In the first experiment we present only the target sentences, without any context. In the second experiment, we first show the context paragraph (three preceding sentences of the target sentence), and ask users to select the most appropriate description of its topic from a list of four candidate topics. Each candidate topic is represented by three words produced by a topic model.[4] Note that the context paragraph consists of original English sentences which did not undergo translation. Once the users have selected the topic, they move to the next screen where they rate the target sentence for naturalness.[5] The third experiment has the same format as the second, except that the three sentences presented prior to rating are randomly sampled from another Wikipedia article.[6] We require annotators to perform a topic identification task prior to rating the target sentence to ensure that they read the context before making acceptability judgments.

For each sentence, we aggregate the ratings from multiple annotators by taking the mean. Henceforth we refer to the mean ratings collected from the first (no context), second (real context), and third (random context) experiments as $\text{H}^{\varnothing}$,

[1] We preprocess the raw dump with WikiExtractor (https://github.com/attardi/wikiextractor), and collect paragraphs that have ≥ 4 sentences with each sentence having ≥ 5 words. Sentences and words are tokenized with spaCy (https://spacy.io/) to check for these constraints.

[2] We use the pre-trained Moses models from http://www.statmt.org/moses/RELEASE-4.0/models/ for translation.

[3] https://www.mturk.com/.

[4] We train a topic model with 50 topics on 15 K Wikipedia documents with Mallet (McCallum, 2002) and infer topics for the context paragraphs based on the trained model.

[5] Note that we do not ask the users to judge the naturalness of the sentence *in context*; the instructions they see for the naturalness rating task is the same as the first experiment.

[6] Sampled sentences are sequential, running sentences.

H^+, and H^-, respectively. We rolled out the experiments on AMT over several weeks and prevented users from doing more than one experiment. Therefore a disjoint group of annotators performed each experiment.

To control for quality, we check that users are rating the English sentences ≥ 3.0 consistently. For the second and third experiments, we also check that users are selecting the topics appropriately. In each HIT one context paragraph has one real topic (from the topic model), and three fake topics with randomly sampled words as the candidate topics. Users who fail to identify the real topic above a confidence level are filtered out. Across the three experiments, over three quarters of workers passed our filtering conditions.

To calibrate for the differences in rating scale between users, we follow the postprocessing procedure of Hill et al. (2015), where we calculate the average rating for each user and the overall average (by taking the mean of all average ratings), and decrease (increase) the ratings of a user by 1.0 if their average rating is greater (smaller) than the overall average by 1.0.[7] To reduce the impact of outliers, for each sentence we also remove ratings that are more than 2 standard deviations away from the mean.[8]

2.2 Results and Discussion

We present scatter plots to compare the mean ratings for the three different contexts ($\text{H}^\varnothing$, H^+, and H^-) in Figure 1. The black line represents the diagonal, and the red line represents the regression line. In general, the mean ratings correlate strongly with each other. Pearson's r for H^+ vs. $\text{H}^\varnothing = 0.940$, H^- vs. $\text{H}^\varnothing = 0.911$, and H^- vs. $\text{H}^+ = 0.891$.

The regression (red) and diagonal (black) lines in H^+ vs. $\text{H}^\varnothing$ (Figure 1a) show a compression effect. Bad sentences appear a little more natural, and perfectly good sentences become slightly less natural, when context is introduced.[9] This is the same compression effect observed by Bernardy et al. (2018). It is also present in the graph for H^- vs. $\text{H}^\varnothing$ (Figure 1b).

Two explanations of the compression effect seem plausible to us. The first is a *discourse coherence* hypothesis that takes this effect to be caused by a general tendency to find infelicitous sentences more natural in context. This hypothesis, however, does not explain why perfectly natural sentences appear less acceptable in context. The second hypothesis is a variant of a *cognitive load* account. In this view, interpreting context imposes a significant burden on a subject's processing resources, and this reduces their focus on the sentence presented for acceptability judgments. At the extreme ends of the rating scale, as they require all subjects to be consistent in order to achieve the minimum/maximum mean rating, the increased cognitive load increases the likelihood of a subject making a mistake. This increases/lowers the mean rating, and creates a compression effect.

The discourse coherence hypothesis would imply that the compression effect should appear with real contexts, but not with random ones, as there is little connection between the target sentence and a random context. By contrast, the cognitive load account predicts that the effect should be present in both types of context, as it depends only on the processing burden imposed by interpreting the context. We see compression in both types of contexts, which suggests that the cognitive load hypothesis is the more likely account.

However, these two hypotheses are not mutually exclusive. It is, in principle, possible that both effects—discourse coherence and cognitive load—are exhibited when context is introduced.

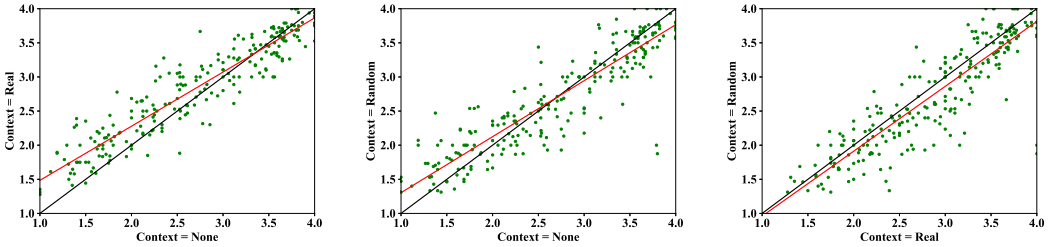
To better understand the impact of discourse coherence, consider Figure 1c, where we compare H^- vs. H^+. Here the regression line is parallel to and below the diagonal, implying that there is a consistent decrease in acceptability ratings from H^+ to H^-. As both ratings are collected with some form of context, the cognitive load confound is removed. What remains is a discourse coherence effect. Sentences presented in relevant contexts undergo a consistent increase in acceptability rating.

To analyze the significance of this effect, we use the non-parametric Wilcoxon signed-rank test (one-tailed) to compare the difference between H^+ and H^-. This gives a p-value of 1.9×10^{-8},

[7] No worker has an average rating that is greater or smaller than the overall average by 2.0.

[8] This postprocessing procedure discarded a total of 504 annotations/ratings (approximately 3.9%) over 3 experiments. The final average number of annotations for a sentence in the first, second, and third experiments is 16.4, 17.8, and 15.3, respectively.

[9] On average, good sentences (ratings ≥ 3.5) observe a rating reduction of 0.08 and bad sentences (ratings ≤ 1.5) an increase of 0.45.



(a) H^{+} vs. $H^{\varnothing}$ (b) H^{-} vs. $H^{\varnothing}$ (c) H^{-} vs. H^{+}

Figure 1: Scatter plots comparing human acceptability ratings.

indicating that the discourse coherence effect is significant.

Returning to Figures 1a and 1b, we can see that (1) the offset of the regression line, and (2) the intersection point of the diagonal and the regression line, is higher in Figure 1a than in Figure 1b. This suggests that there is an increase of ratings, and so, in addition to the cognitive load effect, a discourse coherence effect is also at work in the real context setting.

We performed hypothesis tests to compare the regression lines in Figures 1a and 1b to see if their offsets (constants) and slopes (coefficients) are statistically different.[10] The p-value for the offset is 1.7×10^{-2}, confirming our qualitative observation that there is a significant discourse coherence effect. The p-value for the slope, however, is 3.6×10^{-1}, suggesting that cognitive load compresses the ratings in a consistent way for both H^{+} and H^{-}, relative to $H^{\varnothing}$.

To conclude, our experiments reveal that context induces a cognitive load for human processing, and this has the effect of compressing the acceptability distribution. It moderates the extremes by making very unnatural sentences appear more acceptable, and perfectly natural sentences slightly less acceptable. If the context is relevant to the target sentence, then we also have a discourse coherence effect, where sentences are perceived to be generally more acceptable.

[10] We follow the procedure detailed in `https://statisticsbyjim.com/regression/comparing-regression-lines/` where we collate the data points in Figures 1a and 1b and treat the in-context ratings (H^{+} and H^{-}) as the dependent variable, the out-of-context ratings ($H^{\varnothing}$) as the first independent variable, and the type of the context (real or random) as the second independent variable, to perform regression analyses. The significance of the offset and slope can be measured by interpreting the p-values of the second independent variable, and the interaction between the first and second independent variables, respectively.

3 Modeling Acceptability

In this section, we explore computational models to predict human acceptability ratings. We are interested in models that do not rely on explicit supervision (i.e., we do not want to use the acceptability ratings as labels in the training data). Our motivation here is to understand the extent to which sentence probability, estimated by an unsupervised model, can provide the basis for predicting sentence acceptability.

To this end, we train language models (Section 3.1) using unsupervised objectives (e.g., next word prediction), and use these models to infer the probabilities of our test sentences. To accommodate sentence length and lexical frequency we experiment with several simple normalization methods, converting probabilities to *acceptability measures* (Section 3.2). The acceptability measures are the final output of our models; they are what we use to compare to human acceptability ratings.

3.1 Language Models

Our first model is an LSTM language model (LSTM: Hochreiter and Schmidhuber, 1997; Mikolov et al., 2010). Recurrent neural network models (RNNs) have been shown to be competitive in this task (Lau et al., 2015; Bernardy et al., 2018), and they serve as our baseline.

Our second model is a joint topic and language model (TDLM: Lau et al., 2017a). TDLM combines topic model with language model in a single model, drawing on the idea that the topical context of a sentence can help word prediction in the language model. The topic model is fashioned as an auto-encoder, where the input is the document's word sequence and it is processed by convolutional layers to produce a topic vector to predict the input words. The language model

functions like a standard LSTM model, but it incorporates the topic vector (generated by its document context) into the current hidden state to predict the next word.

We train LSTM and TDLM on 100K uncased English Wikipedia articles containing approximately 40M tokens with a vocabulary of 66K words.[11]

Next we explore transformer-based models, as they have become the benchmark for many NLP tasks in recent years (Vaswani et al., 2017; Devlin et al., 2019; Yang et al., 2019). The transformer models that we use are trained on a much larger corpus, and they are four to five times larger with respect to their model parameters.

Our first transformer is GPT2 (Radford et al., 2019). Given a target word, the input is a sequence of previously seen words, which are then mapped to embeddings (along with their positions) and fed to multiple layers of ''transformer blocks'' before the target word is predicted. Much of its power resides in these transformer blocks: Each provides a multi-headed self-attention unit over all input words, allowing it to capture multiple dependencies between words, while avoiding the need for recurrence. With no need to process a sentence in sequence, the model parallelizes more efficiently, and scales in a way that RNNs cannot.

GPT2 is trained on WebText, which consists of over 8 million web documents, and uses Byte Pair Encoding (BPE: Sennrich et al., 2016) for tokenization (casing preserved). BPE produces sub-word units, a middle ground between word and character, and it provides better coverage for unseen words. We use the released medium-sized model (''Medium'') for our experiments.[12]

Our second transformer is BERT (Devlin et al., 2019). Unlike GPT2, BERT is not a typical language model, in the sense that it has access to both left and right context words when predicting the target word.[13] Hence, it encodes context in a bidirectional manner.

To train BERT, Devlin et al. (2019) propose a masked language model objective, where a random proportion of input words are masked and the model is tasked to predict them based on non-masked words. In addition to this objective, BERT is trained with a next sentence prediction objective, where the input is a pair of sentences, and the model's goal is to predict whether the latter sentence follows the former. This objective is added to provide pre-training for downstream tasks that involve understanding the relationship between a pair of sentences (e.g., machine comprehension and textual entailment).

The bidirectionality of BERT is the core feature that produces its state-of-the-art performance on a number of tasks. The flipside of this encoding style, however, is that BERT lacks the ability to generate left-to-right and compute sentence probability. We discuss how we use BERT to produce a probability estimate for sentences in the next section (Section 3.2).

In our experiments, we use the largest pre-trained model (''BERT-Large''),[14] which has a similar number of parameters (340M) to GPT2. It is trained on Wikipedia and BookCorpus (Zhu et al., 2015), where the latter is a collection of fiction books. Like GPT2, BERT also uses sub-word tokenization (WordPiece). We experiment with two variants of BERT: one trained on cased data (BERT_{CS}), and another on uncased data (BERT_{UCS}). As our test sentences are uncased, a comparison between these two models allows us to gauge the impact of casing in the training data.

Our last transformer model is XLNET (Yang et al., 2019). XLNET is unique in that it applies a novel permutation language model objective, allowing it to capture bidirectional context while preserving key aspects of unidirectional language models (e.g., left-to-right generation).

The permutation language model objective works by first generating a possible permutation (also called ''factorization order'') of a sequence. When predicting a target word in the sequence, the context words that the model has access to are determined by the factorization order. To illustrate this, imagine we have the sequence $\mathbf{x} = [x_1, x_2, x_3, x_4]$. One possible factorization order is: $x_3 \rightarrow x_2 \rightarrow x_4 \rightarrow x_1$. Given this order, if predicting target word x_4, the model only has access to context words $\{x_3, x_2\}$; if the target word is x_2, it sees only $\{x_3\}$. In practice, the target word is set to be the last few words in the factorization

[11] We use Stanford CoreNLP (Manning et al., 2014) to tokenize words and sentences. Rare words are replaced by a special UNK symbol.

[12] https://github.com/openai/gpt-2.

[13] Note that *context* is burdened with two senses in the paper. It can mean the preceding sentences of a target sentence, or the neighbouring words of a target word. The intended sense should be apparent from the usage.

[14] https://github.com/google-research/bert.

Model	Configuration			Training Data			
	Architecture	Encoding	#Param.	Casing	Size	Tokenization	Corpora
LSTM	RNN	Unidir.	60M	Uncased	0.2GB	Word	Wikipedia
TDLM	RNN	Unidir.	80M	Uncased	0.2GB	Word	Wikipedia
GPT2	Transformer	Unidir.	340M	Cased	40GB	BPE	WebText
BERT_{CS}	Transformer	Bidir.	340M	Cased	13GB	WordPiece	Wikipedia, BookCorpus
BERT_{UCS}	Transformer	Bidir.	340M	Uncased	13GB	WordPiece	Wikipedia, BookCorpus
XLNET	Transformer	Hybrid	340M	Cased	126GB	Sentence-Piece	Wikipedia, BookCorpus, Giga5 ClueWeb, Common Crawl

Table 1: Language models and their configurations.

order (e.g., x_4 and x_1), and so the model always sees some context words for prediction.

As XLNET is trained to work with different factorization orders during training, it has experienced both full/bidirectional context and partial/unidirectional context, allowing it to adapt to tasks that have access to full context (e.g., most language understanding tasks), as well as those that do not (e.g., left-to-right generation).

Another innovation of XLNET is that it incorporates the segment recurrence mechanism of Dai et al. (2019). This mechanism is inspired by truncated backpropagation through time used for training RNNs, where the initial state of a sequence is initialized with the final state from the previous sequence. The segment recurrence mechanism works in a similar way, by caching the hidden states of the transformer blocks from the previous sequence, and allowing the current sequence to attend to them during training. This permits XLNET to model long-range dependencies beyond its maximum sequence length.

We use the largest pre-trained model (''XLNet-Large''),[15] which has a similar number of parameters to our BERT and GPT2 models (340M). XLNET is trained on a much larger corpus combining Wikipedia, BookCorpus, news and web articles. For tokenization, XLNET uses SentencePiece (Kudo and Richardson, 2018), another sub-word tokenization technique. Like GPT2, XLNET is trained on cased data.

Table 1 summarizes the language models. In general, the RNN models are orders of magnitude smaller than the transformers in both model parameters and training data, although they are trained on the same domain (Wikipedia), and use uncased data as the test sentences. The RNN models also operate on a word level, whereas the transformers use sub-word units.

[15] https://github.com/zihangdai/xlnet.

3.2 Probability and Acceptability Measure

Given a unidirectional language model, we can infer the probability of a sentence by multiplying the estimated probabilities of each token using previously seen (left) words as context (Bengio et al., 2003):

$$\overrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i}) \quad (1)$$

where s is the sentence, and w_i a token in s.

LSTM, TDLM, and GPT2 are unidirectional models, so they all compute sentence probability as described. XLNET's unique permutational language model objective allows it to compute probability in the same way, and to explicitly mark this we denote it as $\text{XLNET}_{\text{UNI}}$ when we infer sentence probability using only left context words.

BERT is trained with bidirectional context, and as such it is unable to compute left-to-right sentence probability.[16] We therefore compute sentence probability as follows:

$$\overleftrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i}, w_{>i}) \quad (2)$$

With this formulation, we allow BERT to have access to both left and right context words when predicting each target word, since this is consistent with the way in which it was trained. It is important to note, however, that sentence probability computed this way is not a *true probability value*: These probabilities do not sum to 1.0 over all sentences. Equation (1), in contrast, does guarantee true probabilities. Intuitively, the sentence probability computed with this bidirectional formulation is a measure

[16] Technically we can mask all right context words and predict the target words one at a time, but because the model is never trained in this way, we found that it performs poorly in preliminary experiments.

of the model's confidence in the likelihood of the sentence.

To compute the true probability, Wang and Cho (2019) show that we need to sum the pre-softmax weights for each token to score a sentence, and then divide the score by the total score of all sentences. As it is impractical to compute the total score of all sentences (an infinite set), the true sentence probabilities for these bidirectional models are intractable. We use our non-normalized confidence scores as stand-ins for these probabilities.

For XLNET, we also compute sentence probability this way, applying bidirectional context, and we denote it as XLNET_{BI}. Note that $\text{XLNET}_{\text{UNI}}$ and XLNET_{BI} are based on the same trained model. They differ only in how they estimate sentence probability at test time.

Sentence probability (estimated either using unidirectional or bidirectional context) is affected by its length (e.g., longer sentences have lower probabilities), and word frequency (e.g., *the cat is big* vs. *the yak is big*). To modulate for these factors we introduce simple normalization techniques. Table 2 presents five methods to map sentence probabilities to *acceptability measures*: *LP*, *MeanLP*, *PenLP*, *NormLP*, and *SLOR*.

LP is the unnormalized log probability. Both *MeanLP* and *PenLP* are normalized on sentence length, but *PenLP* scales length with an exponent (α) to dampen the impact of large values (Wu et al., 2016; Vaswani et al., 2017). We set $\alpha = 0.8$ in our experiments. *NormLP* normalizes using unigram sentence probability (i.e., $P_{\text{u}}(s) = \prod_{i=0}^{|s|} P(w_i)$), while *SLOR* utilizes both length and unigram probability (Pauls and Klein, 2012).

When computing sentence probability we have the option of including the context paragraph that the human annotators see (Section 2). We use the superscripts $\varnothing$, $+$, $-$ to denote a model using no context, real context, and random context, respectively (e.g., $\text{LSTM}^{\varnothing}$, LSTM^{+}, and LSTM^{-}). Note that these variants are created at test time, and are all based on the same trained model (e.g., LSTM).

For all models except TDLM, incorporating the context paragraph is trivial. We simply prepend it to the target sentence before computing the latter's probability. For TDLM^{+} or TDLM^{-}, the context paragraph is treated as the document context, from which a topic vector is inferred and fed to the language model for next-word prediction. For $\text{TDLM}^{\varnothing}$, we set the topic vector to zeros.

Acc. Measure	Equation
LP	$\log P(s)$
MeanLP	$\frac{\log P(s)}{\lvert s\rvert}$
PenLP	$\frac{\log P(s)}{((5+\lvert s\rvert)/(5+1))^{\alpha}}$
NormLP	$-\frac{\log P(s)}{\log P_{\text{u}}(s)}$
SLOR	$\frac{\log P(s) - \log P_{\text{u}}(s)}{\lvert s\rvert}$

Table 2: Acceptability measures for predicting the acceptability of a sentence; $P(s)$ is the sentence probability, computed using Equation (1) or Equation (2) depending on the model; $P_{\text{u}}(s)$ is the sentence probability estimated by a unigram language model; and α =0.8.

3.3 Implementation

For the transformer models (GPT2, BERT, and XLNET), we use the implementation of *pytorch-transformers*.[17]

XLNET requires a long dummy context prepended to the target sentence for it to compute the sentence probability properly.[18] Other researchers have found a similar problem when using XLNET for generation.[19] We think that this is likely due to XLNET's recurrence mechanism (Section 3.1), where it has access to context from the previous sequence during training.

For TDLM, we use the implementation provided by Lau et al. (2017a),[20] following their optimal hyper-parameter configuration without tuning.

We implement LSTM based on Tensorflow's Penn Treebank language model.[21] In terms of

[17] https://github.com/huggingface/pytorch-transformers. Specifically, we employ the following pre-trained models: `gpt2-medium` for GPT2, `bert-large-cased` for BERT_{CS}, `bert-large-uncased` for BERT_{UCS}, and `xlnet-large-cased` for $\text{XLNET}_{\text{UNI}}$/$\text{XLNET}_{\text{BI}}$.

[18] In the scenario where we include the context paragraph (e.g., $\text{XLNET}^{+}_{\text{UNI}}$), the dummy context is added before it.

[19] https://medium.com/@amanrusia/xlnet-speaks-comparison-to-gpt-2-ea1a4e9ba39e.

[20] https://github.com/jhlau/topically-driven-language-model.

[21] https://github.com/tensorflow/models/blob/master/tutorials/rnn/ptb/ptb_word_lm.py.

hyper-parameters, we follow the configuration of TDLM where applicable. TDLM uses Adam as the optimizer (Kingma and Ba, 2014), but for LSTM we use Adagrad (Duchi et al., 2011), as it produces better development perplexity.

For *NormLP* and *SLOR*, we need to compute $P_u(s)$, the sentence probability based on a unigram language model. As the language models are trained on different corpora, we collect unigram counts based on their original training corpus. That is, for LSTM and TDLM, we use the 100K English Wikipedia corpus. For GPT2, we use an open source implementation that reproduces the original WebText data.[22] For BERT we use the full Wikipedia collection and crawl `smashwords.com` to reproduce BookCorpus.[23] Finally, for XLNET we use the combined set of Wikipedia, WebText, and BookCorpus.[24]

Source code for our experiments is publicly available at: `https://github.com/jhlau/acceptability-prediction-in-context`.

3.4 Results and Discussion

We use Pearson's r to assess how well the models' acceptability measures predict mean human acceptability ratings, following previous studies (Lau et al., 2017b; Bernardy et al., 2018). Recall that for each model (e.g., LSTM), there are three variants with which we infer the sentence probability at test time. These are distinguished by whether we include no context (LSTM$^{\varnothing}$), real context (LSTM^{+}), or random context (LSTM^{-}). There are also three types of human acceptability ratings (ground truth), where sentences are judged with no context, (H$^{\varnothing}$), real context (H^{+}), and random context (H^{-}). We present the full results in Table 3.

To get a sense of what the correlation figures indicate for these models, we compute two human performance estimates to serve as upper bounds on the accuracy of a model. The first upper bound (UB$_1$) is the one-vs-rest annotator correlation, where we select a random annotator's rating and compare it to the mean rating of the rest, using Pearson's r. We repeat this for a large number of trials (1,000) to get a robust estimate of the mean correlation. UB$_1$ can be interpreted as the average human performance working in isolation. The second upper bound (UB$_2$) is the half-vs.-half annotator correlation. For each sentence we randomly split the annotators into two groups, and compare the mean rating between groups, again using Pearson's r and repeating it (1,000 times) to get a robust estimate. UB$_2$ can be taken as the average human performance working collaboratively. Overall, the simulated human performance is fairly consistent over context types (Table 3), for example, UB$_1$ = 0.75, 0.73, and 0.75 for H$^{\varnothing}$, H^{+}, and H^{-}, respectively.

When we postprocess the user ratings, remember that we remove the outlier ratings ($\geq$ 2 standard deviation) for each sentence (Section 2.1). Although this produces a cleaner set of annotations, this filtering step does (artificially) increase the human agreement or upper bound correlations. For completeness we also present upper bound variations where we do not remove the outlier ratings, and denote them as UB$_1^{\varnothing}$ and UB$_2^{\varnothing}$. In this setup, the one-vs.-rest correlations drop to 0.62–0.66 (Table 3). Note that all model performances are reported based on the outlier-filtered ratings, although there are almost no perceivable changes to the performances when they are evaluated on the outlier-preserved ground truth.

Looking at Table 3, the models' performances are fairly consistent over different types of ground truths (H$^{\varnothing}$, H^{+}, and H^{-}). This is perhaps not very surprising, as the correlations among the human ratings for these context types are very high (Section 2).

We now focus on the results with H$^{\varnothing}$ as ground truth (''Rtg'' = H$^{\varnothing}$). *SLOR* is generally the best acceptability measure for unidirectional models, with *NormLP* not far behind (the only exception is GPT2$^{\varnothing}$). The recurrent models (LSTM and TDLM) are very strong compared with the much larger transformer models (GPT2 and XLNET$_{\text{UNI}}$). In fact TDLM has the best performance when context is not considered (TDLM$^{\varnothing}$, *SLOR* = 0.61), suggesting that model architecture may be more important than number of parameters and amount of training data.

For bidirectional models, the unnormalized *LP* works very well. The clear winner here, however,

[22] `https://skylion007.github.io/OpenWebTextCorpus/`.

[23] We use the scripts in `https://github.com/soskek/bookcorpus` to reproduce BookCorpus.

[24] XLNET also uses Giga5 and ClueWeb as part of its training data, but we think that our combined collection is sufficiently large to be representative of the original training data.

Rtg	Encod.	Model	*LP*	*MeanLP*	*PenLP*	*NormLP*	*SLOR*
$H^{\varnothing}$	Unidir.	LSTM$^{\varnothing}$	0.29	0.42	0.42	0.52	**0.53**
		LSTM^{+}	0.30	0.49	0.45	0.61	**0.63**
		TDLM$^{\varnothing}$	0.30	0.49	0.45	0.60	**0.61**
		TDLM^{+}	0.30	0.50	0.45	0.59	**0.60**
		GPT2$^{\varnothing}$	0.33	0.34	**0.56**	0.38	0.38
		GPT2^{+}	0.38	0.59	0.58	**0.63**	0.60
		XLNET$^{\varnothing}_{\text{UNI}}$	0.31	0.42	0.51	0.51	**0.52**
		XLNET$^{+}_{\text{UNI}}$	0.36	0.56	0.55	0.61	**0.61**
	Bidir.	BERT$^{\varnothing}_{\text{CS}}$	0.51	0.54	**0.63**	0.55	0.53
		BERT$^{+}_{\text{CS}}$	0.53	0.63	**0.67**	0.64	0.60
		BERT$^{\varnothing}_{\text{UCS}}$	0.59	0.63	**0.70**	0.63	0.60
		BERT$^{+}_{\text{UCS}}$	0.60	0.68	**0.72**	0.67	0.63
		XLNET$^{\varnothing}_{\text{BI}}$	0.52	0.51	**0.66**	0.53	0.53
		XLNET$^{+}_{\text{BI}}$	0.57	0.65	**0.73**	0.66	0.65
	—	UB_1 / $\text{UB}_1^{\varnothing}$			0.75 / 0.66		
		UB_2 / $\text{UB}_2^{\varnothing}$			0.92 / 0.88		
H^{+}	Unidir.	LSTM$^{\varnothing}$	0.29	0.44	0.43	**0.52**	**0.52**
		LSTM^{+}	0.31	0.51	0.46	**0.62**	**0.62**
		TDLM$^{\varnothing}$	0.30	0.50	0.45	**0.59**	**0.59**
		TDLM^{+}	0.30	0.50	0.46	**0.58**	**0.58**
		GPT2$^{\varnothing}$	0.32	0.33	**0.56**	0.36	0.37
		GPT2^{+}	0.38	0.60	0.59	**0.63**	0.60
		XLNET$^{\varnothing}_{\text{UNI}}$	0.30	0.42	0.50	0.49	**0.51**
		XLNET$^{+}_{\text{UNI}}$	0.35	0.56	0.55	0.60	**0.61**
	Bidir.	BERT$^{\varnothing}_{\text{CS}}$	0.49	0.53	**0.62**	0.54	0.51
		BERT$^{+}_{\text{CS}}$	0.52	0.63	**0.66**	0.63	0.58
		BERT$^{\varnothing}_{\text{CS}}$	0.58	0.63	**0.70**	0.63	0.60
		BERT$^{+}_{\text{CS}}$	0.60	0.68	**0.73**	0.67	0.63
		XLNET$^{\varnothing}_{\text{BI}}$	0.51	0.50	**0.65**	0.52	0.53
		XLNET$^{+}_{\text{BI}}$	0.57	0.65	**0.74**	0.65	0.65
	—	UB_1 / $\text{UB}_1^{\varnothing}$			0.73 / 0.66		
		UB_1 / $\text{UB}_2^{\varnothing}$			0.92 / 0.89		
H^{-}	Unidir.	LSTM$^{\varnothing}$	0.28	0.44	0.43	**0.50**	**0.50**
		LSTM^{-}	0.27	0.41	0.40	**0.47**	**0.47**
		TDLM$^{\varnothing}$	0.29	0.52	0.46	**0.59**	0.58
		TDLM^{-}	0.28	0.49	0.44	**0.56**	0.55
		GPT2$^{\varnothing}$	0.32	0.34	**0.55**	0.35	0.35
		GPT2^{-}	0.30	0.42	**0.51**	0.44	0.41
		XLNET$^{\varnothing}_{\text{UNI}}$	0.30	0.44	**0.51**	0.49	0.49
		XLNET$^{-}_{\text{UNI}}$	0.29	0.40	**0.49**	0.46	0.46
	Bidir.	BERT$^{\varnothing}_{\text{CS}}$	0.48	0.53	**0.62**	0.53	0.49
		BERT$^{-}_{\text{CS}}$	0.49	0.52	**0.61**	0.51	0.47
		BERT$^{\varnothing}_{\text{UCS}}$	0.56	0.61	**0.68**	0.60	0.56
		BERT$^{-}_{\text{UCS}}$	0.56	0.58	**0.66**	0.57	0.53
		XLNET$^{\varnothing}_{\text{BI}}$	0.49	0.48	**0.62**	0.49	0.48
		XLNET$^{-}_{\text{BI}}$	0.50	0.51	**0.64**	0.51	0.50
	—	UB_1 / $\text{UB}_1^{\varnothing}$			0.75 / 0.68		
		UB_2 / $\text{UB}_2^{\varnothing}$			0.92 / 0.88		

Table 3: Modeling results. Boldface indicates optimal performance in each row.

is *PenLP*. It substantially and consistently outperforms all other acceptability measures. The strong performance of *PenLP* that we see here illuminates its popularity in machine translation for beam search decoding (Vaswani et al., 2017). With the exception of *PenLP*, the gain from normalization for the bidirectional models is small, but we don't think this can be attributed to the size of models or training corpora, as the large unidirectional models (GPT2 and $\text{XLNET}_{\text{UNI}}$) still benefit from normalization. The best model without considering context is $\text{BERT}^{\varnothing}_{\text{UCS}}$ with a correlation of 0.70 (*PenLP*), which is very close to the idealized single-annotator performance UB_1 (0.75) and surpasses the unfiltered performance $\text{UB}^{\varnothing}_1$ (0.66), creating a new state-of-the-art for unsupervised acceptability prediction (Lau et al., 2015, 2017b; Bernardy et al., 2018). There is still room to improve, however, relative to the collaborative UB_2 (0.92) or $\text{UB}^{\varnothing}_2$ (0.88) upper bounds.

We next look at the impact of incorporating context at test time for the models (e.g., $\text{LSTM}^{\varnothing}$ vs. LSTM^{+} or $\text{BERT}^{\varnothing}_{\text{UCS}}$ vs. $\text{BERT}^{+}_{\text{UCS}}$). To ease interpretability we will focus on *SLOR* for unidirectional models, and *PenLP* for bidirectional models. Generally, we see that incorporating context always improves correlation, for both cases where we use $\text{H}^{\varnothing}$ and H^{+} as ground truths, suggesting that context is beneficial when it comes to sentence modeling. The only exception is TDLM, where $\text{TDLM}^{\varnothing}$ and TDLM^{+} perform very similarly. Note, however, that context is only beneficial when it is relevant. Incorporating random contexts (e.g., $\text{LSTM}^{\varnothing}$ vs. LSTM^{-} or $\text{BERT}^{\varnothing}_{\text{UCS}}$ vs. $\text{BERT}^{-}_{\text{UCS}}$ with H^{-} as ground truth) reduces the performance for all models.[25]

Recall that our test sentences are uncased (an artefact of Moses, the machine translation system that we use). Whereas the recurrent models are all trained on uncased data, most of the transformer models are trained with cased data. BERT is the only transformer that is pre-trained on both cased (BERT_{CS}) and uncased data (BERT_{UCS}). To understand the impact of casing, we look at the performance of BERT_{CS} and BERT_{UCS} with $\text{H}^{\varnothing}$ as ground truth. We see an improvement of 5–7 points (depending on whether context is incorporated), which suggests that casing has a significant impact on performance. Given that $\text{XLNET}^{+}_{\text{BI}}$ already outperforms $\text{BERT}^{+}_{\text{UCS}}$ (0.73 vs. 0.72), even though $\text{XLNET}^{+}_{\text{BI}}$ is trained with cased data, we conjecture that an uncased XLNET is likely to outperform $\text{BERT}^{\varnothing}_{\text{UCS}}$ when context is not considered.

To summarize, our first important result is the exceptional performance of bidirectional models. It raises the question of whether left-to-right bias is an appropriate assumption for predicting sentence acceptability. One could argue that this result may be due to our experimental setup. Users are presented with the sentence in text, and they have the opportunity to read it multiple times, thereby creating an environment that may simulate bidirectional context. We could test this conjecture by changing the presentation of the sentence, displaying it one word at a time (with older words fading off), or playing an audio version (e.g., via a text-to-speech system). However, these changes will likely introduce other confounds (e.g., prosody), but we believe it is an interesting avenue for future work.

Our second result is more tentative. Our experiments seem to indicate that model architecture is more important than training or model size. We see that TDLM, which is trained on data orders of magnitude smaller and has model parameters four times smaller in size (Table 1), outperforms the large unidirectional transformer models. To establish this conclusion more firmly we will need to rule out the possibility that the relatively good performance of LSTM and TDLM is not due to a cleaner (e.g., lowercased) or more relevant (e.g., Wikipedia) training corpus. With that said, we contend that our findings motivate the construction of better language models, instead of increasing the number of parameters, or the amount of training data. It would be interesting to examine the effect of extending TDLM with a bidirectional objective.

Our final result is that our best model, BERT_{UCS}, attains a human-level performance and achieves a new state-of-the-art performance in the task of unsupervised acceptability prediction. Given this level of accuracy, we expect it would be suitable for tasks like assessing student essays and the quality of machine translations.

[25] There is one exception: $\text{XLNET}^{\varnothing}_{\text{BI}}$ (0.62) vs. $\text{XLNET}^{-}_{\text{BI}}$ (0.64). As we saw previously in Section 3.3, XLNET requires a long dummy context to work, and so this observation is perhaps unsurprising, because it appears that context—whether it is relevant or not—seems to always benefit XLNET.

4 Linguists' Examples

One may argue that our dataset is potentially biased, as round-trip machine translation may introduce particular types of infelicities or unusual features to the sentences (Graham et al., 2019). Lau et al. (2017b) addressed this by creating a dataset where they sample 50 grammatical and 50 ungrammatical sentences from Adger (2003)'s syntax textbook, and run a crowdsourced experiment to collect their user ratings. Lau et al. (2017b) found that their unsupervised language models (e.g., simple recurrent networks) predict the acceptability of these sentences with similar performances, providing evidence that their modeling results are robust.

We test our pre-trained models using this linguist-constructed dataset, and found similar observations: GPT2, BERT_{CS}, and XLNET_{BI} produce a *PenLP* correlation of 0.45, 0.53, and 0.58, respectively. These results indicate that these language models are able to predict the acceptability of these sentences reliably, consistent with our modeling results with round-trip translated sentences (Section 3.4). Although the correlations are generally lower, we want to highlight that these linguists' examples are artificially constructed to illustrate specific syntactic phenomena, and so this constitutes a particularly strong case of out-of-domain prediction. These texts are substantially different in nature from the natural text that the pre-trained language models are trained on (e.g., the linguists' examples are much shorter—less than 7 words on average—than the natural texts).

5 Related Work

Acceptability is closely related to the concept of grammaticality. The latter is a theoretical construction corresponding to syntactic well-formedness, and it is typically interpreted as a binary property (i.e., a sentence is either grammatical or ungrammatical). Acceptability, on the other hand, includes syntactic, semantic, pragmatic, and non-linguistic factors, such as sentence length. It is gradient, rather than binary, in nature (Denison, 2004; Sorace and Keller, 2005; Sprouse, 2007).

Linguists and other theorists of language have traditionally assumed that context affects our perception of both grammaticality (Bolinger, 1968) and acceptability (Bever, 1970), but surprisingly little work investigates this effect systematically, or on a large scale. Most formal linguists rely heavily on the analysis of sentences taken in isolation. However, many linguistic frameworks seek to incorporate aspects of context-dependence. Dynamic theories of semantics (Heim, 1982; Kamp and Reyle, 1993; Groenendijk and Stokhof, 1990) attempt to capture intersentential coreference, binding, and scope phenomena. Dynamic Syntax (Cann et al., 2007) uses incremental tree construction and semantic type projection to render parsing and interpretation discourse dependent. Theories of discourse structure characterize sentence coherence in context through rhetorical relations (Mann and Thompson, 1988; Asher and Lascarides, 2003), or by identifying open questions and common ground (Ginzburg, 2012). While these studies offer valuable insights into a variety of context related linguistic phenomena, much of it takes grammaticality and acceptability to be binary properties. Moreover, it is not formulated in a way that permits fine-grained psychological experiments, or wide coverage computational modeling.

Psycholinguistic work can provide more experimentally grounded approaches. Greenbaum (1976) found that combinations of particular syntactic constructions in context affect human judgments of acceptability, although the small scale of the experiments makes it difficult to draw general conclusions. More recent work investigates related effects, but it tends to focus on very restricted aspects of the phenomenon. For example, Zlogar and Davidson (2018) investigate the influence of context on the acceptability of gestures with speech, focussing on interaction with semantic content and presupposition. The *priming* literature shows that exposure to lexical and syntactic items leads to higher likelihood of their repetition in production (Reitter et al., 2011), and to quicker processing in parsing under certain circumstances (Giavazzi et al., 2018). Frameworks such as ACT-R (Anderson, 1996) explain these effects through the impact of cognitive activation on subsequent processing. Most of these studies suggest that coherent or natural contexts should increase acceptability ratings, given that the linguistic expressions used in processing become more activated. Warner and Glass (1987) show that such syntactic contexts can indeed affect grammaticality judgments in the expected way for

garden path sentences. Cowart (1994) uses comparison between positive and negative contexts, investigating the effect of contexts containing alternative more or less acceptable sentences. But he restricts the test cases to specific pronoun binding phenomena. None of the psycholinguistic work investigates acceptability judgments in real textual contexts, over large numbers of test cases and human subjects.

Some recent computational work explores the relation of acceptability judgments to sentence probabilities. Lau et al. (2015, 2017b) show that the output of unsupervised language models can correlate with human acceptability ratings. Warstadt et al. (2018) treat this as a semi-supervised problem, training a binary classifier on top of a pre-trained sentence encoder to predict acceptability ratings with greater accuracy. Bernardy et al. (2018) explore incorporating context into such models, eliciting human judgments of sentence acceptability when the sentences were presented both in isolation and within a document context. They find a compression effect in the distribution of the human acceptability ratings. Bizzoni and Lappin (2019) observe a similar effect in a paraphrase acceptability task.

One possible explanation for this compression effect is to take it as the expression of cognitive load. Psychological research on the cognitive load effect (Sweller, 1988; Ito et al., 2018; Causse et al., 2016; Park et al., 2013) indicates that performing a secondary task can degrade or distort subjects' performance on a primary task. This could cause judgments to regress towards the mean. However, the experiments of Bernardy et al. (2018) and Bizzoni and Lappin (2019) do not allow us to distinguish this possibility from a coherence or priming effect, as only coherent contexts were considered. Our experimental setup improves on this by introducing a topic identification task and incoherent (random) contexts in order to tease the effects apart.

6 Conclusions and Future Work

We found that processing context induces a cognitive load for humans, which creates a compression effect on the distribution of acceptability ratings. We also showed that if the context is relevant to the sentence, a discourse coherence effect uniformly boosts sentence acceptability. Our language model experiments indicate that bidirectional models achieve better results than unidirectional models. The best bidirectional model performs at a human level, defining a new state-of-the art for this task.

In future work we will explore alternative ways to present sentences for acceptability judgments. We plan to extend TDLM, incorporating a bidirectional objective, as it shows significant promise. It will also be interesting to see if our observations generalize to other languages, and to different sorts of contexts, both linguistic and non-linguistic.

Acknowledgments

We are grateful to three anonymous reviewers for helpful comments on earlier drafts of this paper. Some of the work described here was presented in talks in the seminar of the Centre for Linguistic Theory and Studies in Probability (CLASP), University of Gothenburg, December 2019, and in the Cambridge University Language Technology Seminar, February 2020. We thank the participants of both events for useful discussion.

Lappin's work on the project was supported by grant 2014-39 from the Swedish Research Council, which funds CLASP. Armendariz and Purver were partially supported by the 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 reflect only the authors' views and the Commission is not responsible for any use that may be made of the information it contains.

References

David Adger. 2003. *Core Syntax: A Minimalist Approach*, Oxford University Press, United Kingdom.

John R. Anderson. 1996. ACT: A simple theory of complex cognition. *American Psychologist*, 51:355–365.

Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*, Cambridge University Press.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural

probabilistic language model. *The Journal of Machine Learning Research*, 3:1137–1155.

Jean-Philippe Bernardy, Shalom Lappin, and Jey Han Lau. 2018. The influence of context on sentence acceptability judgements. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pages 456–461. Melbourne, Australia.

Thomas G. Bever. 1970, The cognitive basis for linguistic structures, J. R. Hayes, editor, *Cognition and the Development of Language*, Wiley, New York, pages 279–362.

Yuri Bizzoni and Shalom Lappin. 2019. The effect of context on metaphor paraphrase aptness judgments. In *Proceedings of the 13th International Conference on Computational Semantics - Long Papers*, pages 165–175. Gothenburg, Sweden.

Dwight Bolinger. 1968. Judgments of grammaticality. *Lingua*, 21:34–40.

Ronnie Cann, Ruth Kempson, and Matthew Purver. 2007. Context and well-formedness: the dynamics of ellipsis. *Research on Language and Computation*, 5(3):333–358.

Mickaël Causse, Vsevolod Peysakhovich, and Eve F. Fabre. 2016. High working memory load impairs language processing during a simulated piloting task: An ERP and pupillometry study. *Frontiers in Human Neuroscience*, 10:240.

Wayne Cowart. 1994. Anchoring and grammar effects in judgments of sentence acceptability. *Perceptual and Motor Skills*, 79(3):1171–1182.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. *CoRR*, abs/1901.02860.

David Denison. 2004. *Fuzzy Grammar: A Reader*, Oxford University Press, United Kingdom.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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. Minneapolis, Minnesota.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Maria Giavazzi, Sara Sambin, Ruth de Diego-Balaguer, Lorna Le Stanc, Anne-Catherine Bachoud-Lévi, and Charlotte Jacquemot. 2018. Structural priming in sentence comprehension: A single prime is enough. *PLoS ONE*, 13(4):e0194959.

Jonathan Ginzburg. 2012. *The Interactive Stance: Meaning for Conversation*, Oxford University Press.

Yvette Graham, Barry Haddow, and Philipp Koehn. 2019. Translationese in machine translation evaluation. *CoRR*, abs/1906.09833.

Sidney Greenbaum. 1976. Contextual influence on acceptability judgements. *Linguistics*, 15(187):5–12.

Jeroen Groenendijk and Martin Stokhof. 1990. Dynamic Montague grammar. L. Kalman and L. Polos, editors, In *Proceedings of the 2nd Symposium on Logic and Language*, pages 3–48. Budapest.

Irene Heim. 1982. The Semantics of Definite and Indefinite Noun Phrases. Ph.D. thesis, University of Massachusetts at Amherst.

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

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Aine Ito, Martin Corley, and Martin J. Pickering. 2018. A cognitive load delays predictive eye movements similarly during L1 and L2 comprehension. *Bilingualism: Language and Cognition*, 21(2):251–264.

Hans Kamp and Uwe Reyle. 1993. *From Discourse To Logic*, Kluwer Academic Publishers.

Urvashi Khandelwal, He He, Peng Qi, and Dan Jurafsky. 2018. Sharp nearby, fuzzy far away: How neural language models use context. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 284–294. Association for Computational Linguistics, Melbourne, Australia.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71. Brussels, Belgium.

Jey Han Lau, Timothy Baldwin, and Trevor Cohn. 2017a. Topically driven neural language model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 355–365. Vancouver, Canada.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2015. Unsupervised prediction of acceptability judgements. In *Proceedings of the Joint conference of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2015)*, pages 1618–1628. Beijing, China.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2017b. Grammaticality, Acceptability, and Probability: A Probabilistic View of Linguistic Knowledge. *Cognitive Science*, 41:1202–1241.

William Mann and Sandra Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. `http://mallet.cs.umass.edu`.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association*, pages 1045–1048. Makuhari, Japan.

Hyangsook Park, Jun-Su Kang, Sungmook Choi, and Minho Lee. 2013. Analysis of cognitive load for language processing based on brain activities. In *Neural Information Processing*, pages 561–568. Springer Berlin Heidelberg, Berlin, Heidelberg.

Adam Pauls and Dan Klein. 2012. Large-scale syntactic language modeling with treelets. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 959–968. Jeju Island, Korea.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Daivd Reitter, Frank Keller, and Johanna D. Moore. 2011. A computational cognitive model of syntactic priming. *Cognitive Science*, 35(4):587–637.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725. Berlin, Germany.

Antonella Sorace and Frank Keller. 2005. Gradience in linguistic data. *Lingua*, 115:1497–1524.

Jon Sprouse. 2007. Continuous acceptability, categorical grammaticality, and experimental syntax. *Biolinguistics*, 1123–134.

John Sweller. 1988. Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2):257–285.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36. Association for Computational Linguistics, Minneapolis, Minnesota.

John Warner and Arnold L. Glass. 1987. Context and distance-to-disambiguation effects in ambiguity resolution: Evidence from grammaticality judgments of garden path sentences. *Journal of Memory and Language*, 26(6):714 – 738.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2018. Neural network acceptability judgments. *CoRR*, abs/1805.12471.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27. Washington, DC, USA.

Christina Zlogar and Kathryn Davidson. 2018. Effects of linguistic context on the acceptability of co-speech gestures. *Glossa*, 3(1):73.

Appendix D: CoSimLex: A Resource for Evaluating Graded Word Similarity in Context

CoSimLex: A Resource for Evaluating Graded Word Similarity in Context

Carlos S. Armendariz[*], Matthew Purver[*†], Matej Ulčar[‡], Senja Pollak[†], Nikola Ljubešić[†], Marko Robnik-Šikonja[‡], Mark Granroth-Wilding[◇], Kristiina Vaik[§]

[*]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
[‡]University of Ljubljana, Faculty of Computer and Information Science, Slovenia
{matej.ulcar, marko.robnik}@fri.uni-lj.si
[◇]Department of Computer Science, University of Helsinki, Finland
mark.granroth-wilding@helsinki.fi
[§]Department of Data Analysis, Texta, Estonia
kristiina.vaik@ut.ee

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 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** (Armendariz et al., 2020). 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. The dataset is also multilingual, and includes three less-resourced European languages: Croatian, Finnish and Slovene. It is to be used as the gold standard for evaluation of a 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,

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

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 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**. This effect is well known, and is the basis of many word-sense disambiguation tasks.

1. We finally reached the bank.
2. At this point, the bank was covered with brambles.

In example (1), the word *bank* can have the *financial* or *riverbank* sense; and here, the context doesn't really help us select the correct sense. This creates some tension on the part of the reader: we need to select a sense in order for the sentence to properly work, and without this we may feel that the sentence has not been fully understood. This is an example of *ambiguity*. In example (2), in contrast, the context makes one of the senses more *normal* than the other. Cruse (1986) sees the evaluation of *contextual normality* as the main mechanism for sense selection.
The second way in which context can modify the meaning of a word 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.

3. Sue is visiting her pregnant cousin.
4. Peter doesn't like his cousin.
5. Arthur poured the butter into a dish.

In example (3), the context tells us that the cousin is female. The meaning of *cousin* is being *modulated* by the context to promote the "female" trait. *Cousin* is a *general* word that includes male and female, but also tall, short, happy and sad cousins. However, as we can see in example (4), the absence of information about these traits doesn't produce the type of tension we saw in (1) above; there is a distinction between meaning modulation and sense selection.
The last example (5) is another case of *contextual modulation* in which *poured* highlights 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.
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.
Both *sense selection* and *meaning modulation* 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 becomes something different and follows different rules:

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

Here example (6) 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, example (7), which replaces *dog* with *canine*, should also work, as *canine* could be modulated in the same way that *dog* was; and similarly example (8). However, both seem unnatural at best. 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.
A final interesting point about Cruse's view is that he 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.

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 a more general expression of human cognition (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. We saw an example of the meaning of the word "butter" being *contextually modulated* before, lets see some examples of *salience manipulation* having an effect on the same word:

9. My muffins were a failure, I should have used butter or margarine instead of olive oil.
10. Vegan chefs replace animal fats, like butter, with plant based ones like olive oil or margarine.
11. Vegan influencers believe the consumption of animal products is cruel and unnecessary.

In example (9), in the context of a baking recipe, important dimensions are related to the physical properties of butter, margarine and olive oil. When focusing on these type of dimensions butter and margarine seem more similar because they are both solid while olive oil is liquid. In contrast, in the following example (10) we bring up ideas about veganism and the dimension of animal versus based plant products becomes very salient. This could bring margarine and olive oil closer together and distance both of them from butter, which is an animal product.

There are important differences between this *salience manipulation* effect and the similarly "graded" *contextual modulation* effect. In the previous example (5) *poured* modulated the meaning of the word butter by promoting its "liquid" trait. This effect is limited to the word butter. On the contrary, if the context triggers changes in the salience of conceptual dimensions, any word the annotator evaluates after the change takes place will be affected by it. Once the idea of animal vs plant based is introduced, the change takes place in the mind of the annotator and the perception of the meaning of not only butter, but margarine and olive oil is impacted as well. Our hypothesis is that, by using *salience manipulation*, a context like example (11) can have a impact in the scoring of the similarity of butter, margarine and olive oil without these words even being present in the context. Something that would be impossible if we were looking only at the *contextual modulation* and *sense selection* effects.

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. These results were taken into account when designing 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.

Word1: bank **Word2: money**
Context1
Located downtown along the east **bank** of the Des Moines River ...
Context2
This is the basis of all **money** laundering, a track record of depositing clean money before slipping through dirty money ..

Figure 1: Example from the SCWS dataset, the focus is in the different senses of the word **bank** and there is one independent context per word.

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 contexts did organically occur on their own, this combination of the two did not, 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 effects, 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 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 and Finnish we use existing 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 (Pollak et al., 2020), following the methodology used by Mrkšić et al. (2017) for Croatian.

The English dataset consists of 333 pairs; the Croatian, 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 2 for an example from our English pilot.

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 2: Example from the English pilot, showing a word pair with two contexts, each with mean and standard deviation of human similarity judgements. The original SimLex values for the same word pair without context are shown for comparison.

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). As explained in their original paper, *cup* and *mug* are very similar, while *coffee* and *cup* are strongly related but not similar at all. For English we adopted a modified version of their crowd-sourcing process: we use *Amazon Mechanical Turk*, with the same scoring scale (0 to 6), 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, Finnish 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, Finnish and Slovene). 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.

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 annota-

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

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 younger 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. He decides to marry Belinda and take her **child** with him.)	

Figure 3: Example from the Croatian pilot, showing the word pair with two contexts, each with 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*. English translations (generated using Google Translate with manual post-correction) are shown here for exposition purposes but are not part of the dataset.

tion. 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 rating 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, 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.

Word1	Word2	Context1	Context2	STDev1	STDev2	P-Value
water	ice	2.57	8.13	2.60	1.82	2.18E-08**
friendly	generous	4.44	3.92	2.85	3.56	0.258
keep	protect	2.50	3.75	2.66	2.22	0.036**
pact	agreement	8.73	8.97	1.89	1.53	0.302
narrow	broad	0.42	1.97	1.19	2.60	0.012**
arm	neck	3.81	1.27	2.89	1.97	0.002**
cottage	cabin	8.07	9.56	2.37	0.94	0.003**
inform	notify	9.31	9.80	0.97	0.55	0.019**
mother	guardian	3.94	7.28	3.15	2.54	0.0001**
car	bicycle	4.12	4.85	2.58	2.46	0.169

Table 1: Example results from the English dataset showing the mean similarity, standard deviation and p-value calculated using the Mann-Whitney U test.

Word1	Word2	Context1	Context2	STDev1	STDev2	P-Value
nadbiskup	biskup	6.67	6.30	2.66	2.61	0.325
sretan	mlad	1.50	0.30	2.28	0.67	0.099*
kost	čeljust	6.00	3.33	2.38	2.24	0.013**
zvijer	životinja	9.44	6.30	1.09	3.09	0.004**
priča	tema	2.59	7.64	1.88	2.51	0.0004**

Table 2: Example results from the Croatian dataset showing the mean similarity, standard deviation and p-value calculated using the Mann-Whitney U test.

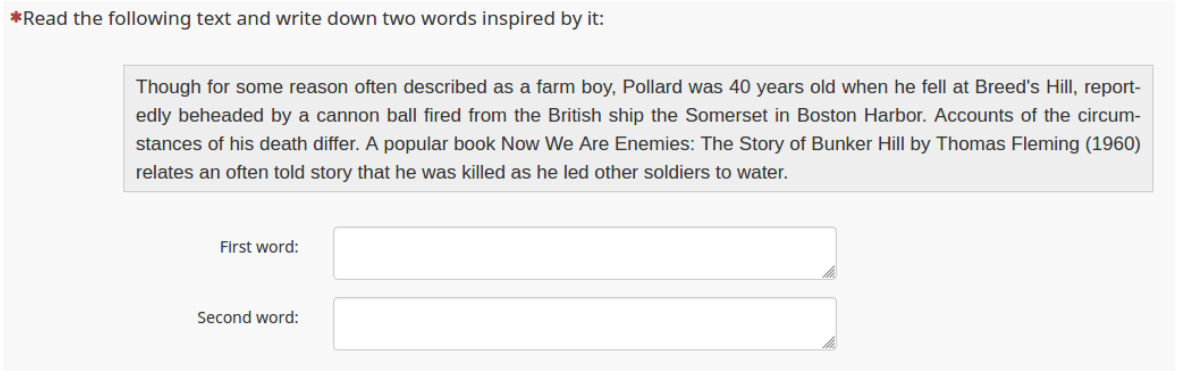


Figure 4: First page shown for each word pair annotation task: annotators must read the context and come up with two words inspired by it. At this point, the word pair to be scored is not known to the annotator.

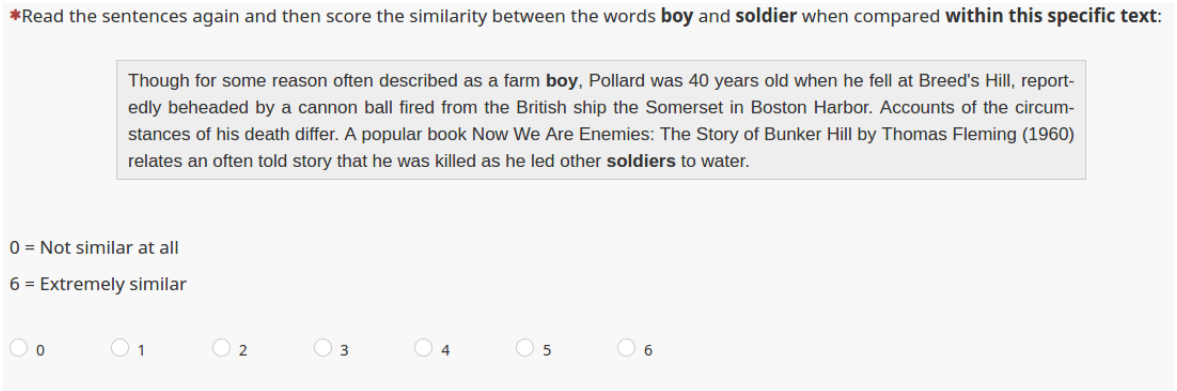


Figure 5: Second page shown for each word pair annotation task: the same context is now shown with the target words in bold, and annotators must give a similarity score for the word pair within that particular context.

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 dataset contains 341 entries in English, 113 in Croatian, 112 in Slovene and 25 in Finnish. Each of the entries contains a pair of words evaluated in two different contexts. Please see Table 1 for examples from the English results and Table 2 for Croatian.

Inter-rater agreement (IRA) is measured as Spearman correlation between each rating and the average values. After post-processing the data our dataset's IRA is surprisingly similar between the different languages: for English, Croatian and Slovene the mean is $\rho = 0.77$, while Finnish achieves $\rho = 0.80$ (likely due to the small sample); these compare well to other related datasets (SimLex-999 $\rho = 0.78$, SCWS $\rho = 0.52$). The crowd-sourced nature of English data results in a higher percentage of annotations being dropped in the post-processing. However the fact that both methods converge to the same IRA is encouraging, and seems to indicate that both methods achieve comparable results.

The statistical significance of the difference in similarity evaluation between contexts was assessed using the Mann-Whitney U test. In English, from the 341 entries, 220 results showed a statistically significant difference at $p < 0.1$ (ratio = 0.65), and 208 did so at $p < 0.05$ (ratio = 0.61). The results again where quite similar for Croatian and Slovene with 73 and 65 statistically significant results at $p < 0.1$ (ratio = 0.65 and 0.58). Finnish results showed a much smaller ratio of statistically significant results (8 entries, ratio = 0.33), which could be due to the small sample but may deserve further investigation. However, as in the case of the inter-rater agreement, the fact that English, Croatian and Slovene results are very similar is a good sign for both methodologies: the English crowd-sourced annotation and the smaller sample of better quality annotation we used for Croatian and Slovene.

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 was provided for the evaluation stage of SemEval 2020 at the beginning of February 2020, and will be made public when the competition is over (before the LREC2020 conference).

7. Acknowledgements

This research is supported by the 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 reflect only the authors' views and the Commission is not responsible for any use that may be made of the information it contains. The first author is also supported by the EPSRC and AHRC Centre for Doctoral Training in Media and Arts Technology (EP/L01632X/1); the second author is also supported by the EPSRC project Streamlining Social Decision Making for Improved Internet Standards (SoDeStream, EP/S033564/1).

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 in Linköping Electronic Conference Proceedings, 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.

9. Language Resource References

Armendariz, Carlos Santos and Purver, Matthew and Ulčar, Matej and Pollak, Senja and Ljubešić, Nikola and Robnik-Šikonja, Marko and Granroth-Wilding, Mark and Vaik, Kristiina. (2020). *CoSimLex*. EMBEDDIA project, ISLRN 613-489-674-355-0. Available via CLARIN `http://hdl.handle.net/11356/1308`.

Pollak, Senja and Ljubešić, Nikola and Vulić, Ivan. (2020). *A Slovenian Translation for SimLex*. EMBEDDIA project, ISLRN 613-489-674-355-0. Available via CLARIN `http://hdl.handle.net/11356/1309`.

Appendix E: SemEval-2020 Task 3: Graded Word Similarity in Context

SemEval-2020 Task 3: Graded Word Similarity in Context

Carlos S. Armendariz[*] and **Matthew Purver**[*†]
[*]Cognitive Science Research Group, Queen Mary University of London, London, UK
{c.santosarmendariz, m.purver}@qmul.ac.uk

Senja Pollak[†] and **Nikola Ljubešić**[†]
[†]Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
{senja.pollak, nikola.ljubesic}@ijs.si

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

Ivan Vulić
University of Cambridge, UK
iv250@cam.ac.uk

Mohammed Taher Pilehvar
Tehran Institute for Advanced Studies
mp792@cam.ac.uk

Abstract

This paper presents the *Graded Word Similarity in Context (GWSC)* task which asked participants to predict the effects of context on human perception of similarity in English, Croatian, Slovene and Finnish. We received 15 submissions and 11 system description papers. A new dataset (CoSimLex) was created for evaluation in this task: it contains pairs of words, each annotated within two short text passages. Systems beat the baselines by significant margins, but few did well in more than one language or subtask. Almost every system employed a Transformer model, but with many variations in the details: WordNet sense embeddings, translation of contexts, TF-IDF weightings, and the automatic creation of datasets for fine-tuning were all used to good effect.

1 Introduction

Contextualised word embeddings, produced by models such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019), have quickly become the standard in NLP systems. They deliver impressive performance in language modeling and downstream tasks; but there are few resources available which allow intrinsic evaluation in terms of the properties of the embeddings themselves, or their ability to model human perception of meaning, and how these depend on context. For non-contextualised models, resources like WordSim-353 (Finkelstein et al., 2002) and SimLex-999 (Hill et al., 2015) were instrumental to evaluate their ability to reflect human similarity judgements. However these datasets treat pairs of words in isolation, and thus cannot tell us much about the effect of context. The few resources that work with context, like SCWS (Huang et al., 2012), WiC (Pilehvar and Camacho-Collados, 2019), and WSim (Erk et al., 2013), focus on word sense and discrete effects, thus missing the more graded effects that context has on words in general, and that approaches like ELMo and BERT would seem well suited to model. Further, USim (Erk et al., 2013) focuses on separate sentential contexts only in the English language.

The goal of SemEval-2020 Task 3: Graded Word Similarity in Context, was to move towards filling that gap. We created a new dataset, **CoSimLex** (Armendariz et al., 2020), which builds on the familiar pairwise, graded similarity task of SimLex-999, but extends it to pairs of words as they occur in context; specifically, each pair of words appears together in two different shared contexts (see Figure 1). The task was designed to test the ability of participating systems to reflect human judgements of word meaning similarity in context, and crucially, the way in which this varies as context is changed. In addition, since CoSimLex takes the *gradedness* of human judgements into account, the task applies not only to polysemous words, or words with distinct senses, but to the phenomenon of context-dependency of word

meaning in general. The dataset is also multi-lingual: besides English, it includes three less-resourced European languages, Croatian, Finnish, and Slovene.

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

Figure 1: Example from the English dataset, showing a word pair with two contexts, each with mean and standard deviation of human similarity judgements. The original SimLex values for the same word pair without context are shown for comparison. The P-Value shown is the result of a Mann-Whitney U test.

2 Background

Our motivation lies in the cognitive and psychological mechanisms by which context affects our perception of word meaning. Here, we present two of the most prominent ideas that helped define the task and dataset, and explain why previous datasets for similarity in context are not well suited to test them.

2.1 Contextual Modulation

One debate in lexical semantics is whether the discreteness of lexical senses is fundamental or just a perception. Cruse (1986) proposed a compromise, distinguishing two different manners in which sentential context modifies the meaning of a word. First, the context can select for different discrete senses; in this case, the word is described as *ambiguous*, and the process as **contextual selection of senses** (familiar from many word sense disambiguation tasks). Second, the context can modify meaning within the scope of a single sense by *highlighting* certain semantic traits and *backgrounding* others. This is described as **contextual modulation of meaning**, and the word as *general* with respect to the traits being modulated. This latter effect is not discrete, but continuous or graded; every word is *general* to some extent, and thus has a different meaning in every context in which it appears.

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

The main effect of the context in example (1) is to *select* one of the discrete senses associated with the word *bank*. In contrast, in examples (2) and (3), the contexts *modulate* the meanings of the words *cousin* and *butter*: for *cousin*, promoting the "female" trait, and for *butter*, the "liquid" trait. This is possible because of the *general* quality of these words. Other traits could be promoted in different contexts: *cousin* includes male and female, but also tall, short, happy and sad cousins. Related traits can be promoted as a consequence of this modulation: we understand the butter as not only liquid, but warm. We expect this to affect similarity judgements.

2.2 Salience Manipulation

In contrast to this purely linguistic view, we can take a cognitive perspective on language and meaning, seeing it as a more general expression of human cognition (Evans and Green, 2018). In this view, the units of interest are the *conceptual structures* associated with words or lexical units, rather than the words themselves. One approach is to see these in terms of *conceptual spaces* characterised by *quality dimensions* (Gärdenfors, 2000; Gärdenfors, 2014). These dimensions may be concrete (weight,

temperature, brightness) or abstract (awkwardness, goodness), and concepts are defined as regions (usually convex) within the space. This space is not fixed: when we communicate we constantly re-negotiate the dimensions framing the conversation and their salience (Warglien and Gärdenfors, 2015). This **salience manipulation** changes their perceived importance. Priming effects are proposed as the main mechanism that facilitates this process (Pickering and Garrod, 2004). This type of semantic effect was first reported by Meyer and Schvaneveldt (1971) when they found that their lexical decision task was responded to faster when the subjects were primed with words associated to the target words.

From this perspective, then, context affects meaning not via the presence of specific words, but via a change in the *mental state* of the hearer/reader.

1. My muffins were a failure, I should have used butter or margarine instead of olive oil.
2. Vegan chefs replace animal fats, like butter, with plant based ones like olive oil or margarine.
3. Vegan influencers believe the consumption of animal products is cruel and unnecessary.

In example (1), the context of baking increases the salience of dimensions related to physical properties of ingredients; butter and margarine (both solid) therefore seem more similar to each other than to olive oil (liquid). In contrast, example (2)'s context of veganism makes the animal vs. plant-based dimension very salient; margarine and olive oil now seem more similar to each other than to the animal-based butter.

The effects of *salience manipulation* and *contextual modulation* have important differences. The effect in example (3) is introduced by the word *poured* and limited to the word *butter*, but the effect in example (1) seems more general: once a context triggers changes in the salience of conceptual dimensions, any word thereafter is affected. Our hypothesis is that the *salience manipulation* effect applies even when the target words are not present: a context like example (3) will impact later perceptions of similarity of butter, margarine and olive oil. We hope to test such predictions in later analyses.

2.3 Related Work

The **Stanford Contextual Word Similarity (SCWS)** dataset (Huang et al., 2012), and the similar **USim** dataset (Erk et al., 2013) contain graded similarity judgements of pairs of words in the context of naturally occurring sentences (e.g., from Wikipedia with SCWS). However, the datasets were designed to evaluate a discrete multi-prototype model, so the focus was on contexts that select for discrete word senses, and each word in a pair was presented in its own distinct context. This prevents a systematic comparison of contextual effects on pairwise similarity. In addition, inter-rater agreement (IRA) on SCWS, measured as the Spearman correlation between different annotators, shows worryingly low scores. As Pilehvar and Camacho-Collados (2019) point out, the mean IRA between each annotator and the average of the rest, considered a human-level upper bound for model performance, is 0.52; while the performance of a simple context-independent model like word2vec (Mikolov et al., 2013) is 0.65. Many scores also show a very large standard deviation, with annotators rating the same pair very differently. One possible reason may lie in the annotation design: the task itself does not directly enforce engagement with the context, and the target words were presented to annotators highlighted in boldface, making it easy to pick them out from the context without reading it.

Some 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, and labelled as to whether the word sense in the two examples/contexts is the same or different. This forces engagement with the context; it also creates a task in which context-independent models like word2vec "would perform no better than a random baseline"; and inter-rater agreement scores are much more healthy. However, as the dataset focuses on discrete word senses, it cannot capture graded effects of context.

These datasets are also available only in English. Multi-lingual similarity datasets exist: in **SemEval-2017 Task 2: Multilingual and Cross-lingual Semantic Word Similarity**, Camacho-Collados et al. (2017) used five different languages, and even used pairs in which each word was presented in a different language. A more recent **Multi-SimLex** dataset (Vulić et al., 2020) comprises similarity ratings for 1,888 concept pairs aligned across 13 typologically diverse languages. However, the pairs in both datasets were annotated out of context, preventing analysis of contextual effects.

3 Task Description

Our dataset is based on pairs of words from SimLex-999 (Hill et al., 2015). Each instance is a naturally-occurring context, taken from Wikipedia, in which both words in the pair appear, labelled with a similarity score given by human annotators. For each pair, the dataset contains two different contexts (see Section 4 for more detail on dataset and choice of contexts). We proposed two different subtasks: first, to predict the change in similarity score between the two different contexts for each pair; second, to predict the similarity scores themselves. These are related but independent tasks that use the same input data, but each subtask had its own phases and leaderboards. Submissions for each subtask were independent and participants were able to use different models for each subtasks and each language. The tasks were unsupervised, and so no training data was released; However, we released a small *practice kit* which contained a practice dataset, a script to generate the baseline and evaluation scripts so participants could easily reproduce results, and understand how the dataset looked and how the task was evaluated.

3.1 Subtask 1: Predicting Change

In the first subtask, participants were asked to predict the *change* in the similarity ratings of a pair of words when the human annotators are presented with the same word pair within two different contexts. This task directly addresses our main question. It evaluates how well systems are able to model the effect that context has in human perception of similarity. Theoretically a model could perform very well at modelling change without actually being able to accurately predict the ratings themselves. On the other hand, any context-independent model will predict no change and perform poorly in this task.

3.2 Subtask 2: Predicting Contextual Ratings

In the second subtask, participants were asked to predict the absolute similarity rating for each pair in each context. This is a more traditional task which evaluates systems' ability to model both similarity of words and the effect that context has on it. Good context-independent models could theoretically give reasonably competitive results in this task, however we still expect context-dependent models to have a considerable advantage.

4 Dataset

CoSimLex (Armendariz et al., 2020) is based on pairs of words from SimLex-999 (Hill et al., 2015); the reliability and common use of SimLex makes it a good starting point and allows comparison of judgements and model outputs to the context-independent case. For Croatian and Finnish we use existing 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,[1] following Mrkšić et al. (2017)'s methodology for Croatian.

The dataset consists of 340 pairs in English, 112 in Croatian, 111 in Slovene and 24 in Finnish. Each pair is rated within two different contexts, giving a total of 1174 scores of contextual similarity. This poses a difficult task: to find suitable, organically occurring contexts; this task is even more challenging for languages with less resources, and as a result the selection of pairs is different for each language.

Each line of CoSimLex is 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, calculated as the mean of annotator ratings for that context; two scores of standard deviation; the p-value given by applying the Mann-Whitney U test to the two score distributions; and the four inflected forms of the words exactly as they appear in the contexts (including case; note that in the morphologically rich languages, many inflections are possible). To the best of our knowledge, this is the first reasonably sized dataset in which differences in contextual similarity between two words are supported with a test of statistical significance. Figure 1 shows an example from the English dataset.

4.1 Context Selection

For each word pair we needed to find two suitable contexts. These contexts were extracted from each language's Wikipedia. They are made of three consecutive sentences and they needed to contain the pair

[1] Available from `http://hdl.handle.net/11356/1309`

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 three languages are highly inflected (Croatian, Finnish and Slovene). To overcome this, we worked with data from (Ginter et al., 2017)[2] which contains tokenised and lemmatised versions of Wikipedia for 45 languages.

The differences were expected to be small; to maximise the chance of finding contexts that produced different ratings of similarity, we used a dual process based on ELMo and BERT models. First, we used a model to rate the similarity between the target words within each of the candidate contexts; then selected the context in which it scored the pair as the most similar, and the context in which it scored them as most different. We repeated the process using both ELMo and BERT scores. This gave us 4 promising contexts. Then we added 4 randomly selected contexts for a total of 8 candidate contexts.

The final selection of two contexts was made by expert human annotators, one per language. Our experts were presented with 8 candidate contexts and asked to select the two that maximised the potential contrast in similarity. In the case of less-resourced languages, the smaller size and lower quality of the Wikipedia text resources required some extra steps to ensure the quality of the final annotation. A set of heuristic filters were used to try to remove badly constructed contexts. In addition we produce 16 candidates instead of 8 for the expert annotators to choose from.

4.2 Annotation

As starting point for our annotation methodology, we adapted the instructions used for SimLex-999. This way we benefited from its tested method of explaining how to focus on *similarity* rather than *relatedness* or *association* (Hill et al., 2015). As explained in their original paper, *cup* and *mug* are very similar, while *coffee* and *cup* are strongly related but not similar at all. For English we adopted their crowd-sourcing process: we used *Amazon Mechanical Turk*, with the same initial scoring scale (0 to 6), which is later transformed to a 0 to 10 scale. For the less-resourced languages, crowdsourcing is not a viable option due to lack of available speakers, and we recruited annotators directly. This means fewer annotators (for Croatian, Finnish and Slovene, 12 annotators vs 27 in English), however the average quality of annotation is higher and the data requires less post-processing.

In regards to the annotation process itself, 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 defined three goals:

- Interaction with the context should be as natural as possible, so as to maximise priming effects and capture the potential change in the salience of conceptual dimensions.
- Annotators should have the chance to account for lexical modulation within the sentence.
- The process should ensure that the annotators engage fully with the context.

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 target words now highlighted in bold; they are now asked to rate the similarity of target words within the sentence. Notice these target words are completely independent to the ones that were chosen as "inspired by the context" (see Apendix A for an example of the survey).

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 previous 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

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

Dataset	#pairs	Sim	StDev	Spearman's ρ	Change (Abs)	$p < 0.1$	$p < 0.05$
SimLex-999	999	4.56	1.27	0.78	-	-	-
English CoSimLex	340	5.54	2.24	0.77	2.16	65%	61%
Croatian CoSimLex	112	4.39	2.23	0.76	2.32	65%	54%
Slovene CoSimLex	111	4.90	2.17	0.77	1.96	59%	46%
Finnish CoSimLex	24	4.08	2.16	0.81	1.75	33%	29%

Table 1: Similarity, standard deviation, Spearman's ρ and change are average values. The two rightmost columns denote the proportion of pairs whose differences of scores with the original values are statistically significant at p-value < 0.1 and p-value < 0.05.

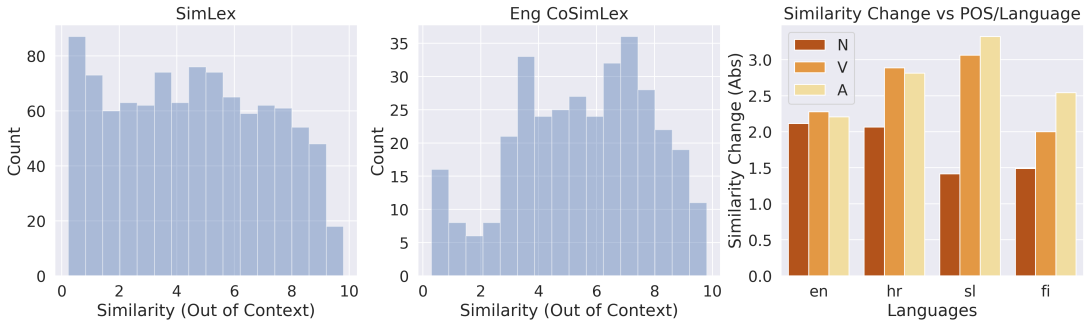


Figure 2: (a) (b): Differences in the distribution of similarity between SimLex-999 and the English CoSimLex; **(c):** Change in the scoring of similarity between contexts categorized by language and part of speech

to not read the rest of the contexts. Our initial task is designed to prevent this (the words are not in bold in the first page).

4.3 Post-Processing

Post-processing and cleaning the data is especially important when relying on crowd-sourcing platforms to source annotators. Reliability of annotation was ensured by an adapted version of SimLex-999's post-processing method, which includes rating calibration and the filtering of annotators with very low correlation to the rest, see the original paper for details (Hill et al., 2015). In addition, we were able to use responses to the first annotation question to check annotator engagement with the context.

In English there were instances in which a block of annotations resulted in especially bad data. In those cases the only solution was repeating the annotation of the whole block. In our experience, obtaining good annotation using Amazon Mechanical Turk is not straightforward, but can be improved by a few strategies to attract good annotators. It is possible to engage with quality annotators and create private tasks for them inside the platform, which produces better data and allows higher payment for the worker. We encourage other researchers to use similar strategies when possible. This was not an issue with the rest of the languages, where annotators were sourced directly. After the post-processing steps the English dataset retained an average of 21 annotations per entry (from a starting point of 27) while the rest of the languages kept an average of 10 annotations (from the starting 12).

4.4 Basic Analysis

The difficulty of finding contexts for the less-resourced languages restricted the selection of pairs available. As a consequence the overlap of pairs between different languages is smaller than originally intended (86 pairs appear in two languages, 12 in three and only 4 appear in all languages). However we were still able to replicate SimLex-999's proportions of nouns, verbs and adjectives (about two thirds nouns, two ninths verbs and one ninth adjectives). In English we checked other metrics, namely concreteness, standard deviation and out-of-context similarity. The first were kept in similar ranges to SimLex, however for out-of-context similarity we decided to lower the proportion of antonyms and low similarity score pairs, which as noted by Camacho-Collados et al. (2017) were substantially overrepresented (see Figure 2).

We expected that the relative complexity of the annotation process and the increased confounding effects could affect inter-rater agreement; however, as we can see in Table 1, the different CoSimLex datasets show correlation scores very close to SimLex-999's IRA (ρ = 0.77 vs ρ = 0.78 in English). In the same table we can see the standard deviation is higher. Differences in the average similarity score are mainly due to the pair selection. After the post-processing and cleaning of the data both the crowdsourced and directly sourced annotation produced similar IRA and standard deviation. We wondered if the highly inflected nature of some of the languages might increase the contextual effects; but as can be seen in the table, the average change is very similar, even lower for Slovene and Finnish. However an interesting phenomenon seems to appear when we look at the distribution by part of speech; Chart (c) in Figure 2 suggest that verbs and adjectives in Croatian, Slovene and Finnish do see an increased effect of context compared with English ones. Importantly, the global percentage of statistically significant results is high (indeed, higher than we expected), with a global 62% of pairs showing statistically significant differences between contexts.

One potential confounding effect is the separation between words as presented in context (the number of intervening words between the target pair): it is possible this could affect annotators' perception of similarity. There is a very small negative correlation between similarity ratings and distance (Pearson r = -0.13). The source of this could be annotator bias, a linguistic effect or a combination of the two; but the effect seems small enough to ignore for current purposes.

5 Evaluation Metrics

The first subtask looked at the change in similarity between the two contexts, therefore it was important to preserve the difference between positive and negative values since it reflected in which of the two context the system believed the two words to be more or less similar. Consequently the most appropriate metric was **Uncentered Pearson Correlation** which looks at the deviation from zero instead of the mean.

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

For the second subtask, which looked at the more traditional absolute value of similarity in context, we followed (Camacho-Collados et al., 2017) and used the harmonic mean of the Pearson and the Spearman correlations between the system's results and the average of the human annotations.

6 Baselines

Our task studies contextual effects in four different languages, which made Multiligual BERT the perfect candidate for our baseline. Released shortly after the original BERT model (Devlin et al., 2019), it employs its same architecture while being trained in more than 100 different languages, our four languages between them. The original model introduced an innovative masking strategy that for the first time allowed for a bidirectional Transformer language model. BERT models are renowned for their ability to capture contextual effects, ability which is often blamed for an important part of their performance improvements. For the baseline of our task we used the uncased version of the model, and as a common strategy we used the contents of the last layer to form our embeddings. BERT creates sub-word tokens for the out of vocabulary words, in those cases our strategy was simply averaging the sub-word vectors to form a word embeddings.

Additionally, the results achieved by ELMo are added to Tables 2 and 3 as a reference. This model precedes BERT and was one of the first to produce contextualised embeddings (Peters et al., 2018), in this case using a bidirectional LSTM. The original ELMo dataset was only trained in English, however we used ELMo models recently trained in Croatian, Slovene and Finnish (Ulčar and Robnik-Šikonja, 2020).

7 Participants & Results

The task received a total of 14 submissions for the first subtask and 15 submissions for the second. From those, 11 teams submitted system description papers for review. In order to be considered for the official rankings we asked participants to fill a form with some basic information about their systems. Teams that

SUBTASK 1							
English		Croatian		Slovene		Finnish	
Ferryman	0.774	BabelEnconding	0.74	Hitachi	0.654	will_go	0.772
will_go	0.768	Hitachi	0.681	BRUMS	0.648	Ferryman	0.745
MultiSem	0.76	BRUMS	0.664	BabelEnconding	0.646	BabelEnconding	0.726
LMMS	0.754	Ferryman	0.634	CiTIUS-NLP	0.624	BRUMS	0.671
BRUMS	0.754	LMMS	0.616	Ferryman	0.606	CiTIUS-NLP	0.671
Hitachi	0.749	will_go	0.597	will_go	0.603	MultiSem	0.593
BabelEnconding	0.73	CiTIUS-NLP	0.587	LMMS	0.56	Hitachi	0.574
CiTIUS-NLP	0.721	MineriaUNAM	0.374	MineriaUNAM	0.328	MineriaUNAM	0.389
MineriaUNAM	0.544	MultiSem	-	MultiSem	-	LMMS	0.36
JUSTMasters	0.738		0.44		0.512		0.546
UZH	0.765		-		-		-
mBERT_uncased	0.713		0.587		0.603		0.671
ELMo	0.570		0.662		0.452		0.550

Table 2: Subtask 1 Final Ranking: The values are calculated as the Pearson Uncentered Correlation between the system's scores and the average human annotation. It represents the system's ability to predict the change in perception produced by the contexts. Since different annotators looked at each context, human performance couldn't be calculated for this subtask. JUSTMasters and UZH are not part of the official ranking since they were able to optimise their systems with more than the competition's limit of 9 submissions.

SUBTASK 2							
English		Croatian		Slovene		Finnish	
MineriaUNAM	0.723	BabelEnconding	0.658	BabelEnconding	0.579	BRUMS	0.645
LMMS	0.72	Hitachi	0.616	BRUMS	0.573	BabelEnconding	0.611
AlexU-Aux-Bert	0.719	MineriaUNAM	0.613	CiTIUS-NLP	0.538	MineriaUNAM	0.597
MultiSem	0.718	LMMS	0.565	will_go	0.516	MultiSem	0.492
BRUMS	0.715	BRUMS	0.545	AlexU-Aux-Bert	0.516	Ferryman	0.357
will_go	0.695	CiTIUS-NLP	0.496	Hitachi	0.514	LMMS	0.354
Hitachi	0.695	AlexU-Aux-Bert	0.402	MineriaUNAM	0.487	will_go	0.35
CiTIUS-NLP	0.687	will_go	0.402	LMMS	0.483	Hitachi	0.335
BabelEnconding	0.634	Ferryman	0.397	Ferryman	0.345	CiTIUS-NLP	0.289
Ferryman	0.437	MultiSem	-	MultiSem	-	AlexU-Aux-Bert	0.289
JUSTMasters	0.725		0.443		0.44		0.68
mBERT_uncased	0.573		0.402		0.516		0.289
ELMo	0.510		0.529		0.407		0.516
Human	0.77		0.76		0.77		0.81

Table 3: Subtask 2 Final Ranking: The values are calculated as the harmonic mean of the Spearman and Pearson correlation between the system's scores and the average human annotation. It represents the system's ability to predict contextual human perception of similarity. Human performance is the average value when comparing each annotator against the average of the rest. JUSTMasters is not part of the official ranking since they were able to optimise their system with more than the competition's limit of 9 submissions.

neither filled the form nor submitted a system description paper do not appear in the official rankings (Tables 2 and 3). We will discuss here the results of the remaining 11 systems.

First, we describe a group of systems designed around sense embeddings created using WordNet (Miller, 1995) as a guide. The most successful was the submission by **LMMS**. They employed a similar strategy to the one set out in (Loureiro and Jorge, 2019), creating pretrained embeddings for each sense in WordNet, this time using XLM-R (Conneau et al., 2019) and SemCor augmented with their own UWA dataset (Loureiro and Camacho-Collados, 2020). This approach achieved second place in the English Subtask 1 and fourth in the English Subtask 2. **UZH** (Tang, 2020) submitted (after the competition had ended) a system based on the original BERT sense embeddings created for (Loureiro and Jorge, 2019) but improved their performance by combining them with contextualised embeddings. Finally for this group **AlexU-AUX-BERT** (Mahmoud and Torki, 2020) created new sense embeddings for the competition

target words. In order to do so they sourced additional contexts for the top WordNet synsets. Their system scored third in the English Subtask 2. The pretrained WordNet sense embedding proved highly successful in this task, especially in Subtask 2, predicting the similarity scores themselves. The biggest weakness of the approach is their reliance on linguistic resources that don't exist for most languages other than English.

Related to these systems, the submission by **MineriaUNAM** (Gomez-Adorno et al., 2020) won the English Subtask 2. They proposed a system in which they calculated K-Means inspired centroids from the words in the context and used them to modify the original SimLex-999 non contextualised similarity scores. The approach, even if very successful, seems to rely on having out of context human annotations, perhaps not realistic in the general case. The fact that the system did very poorly in Subtask 1, which asked to predict change, seems to indicate much of the success is coming from the human annotations. A related strategy could perhaps be used with embeddings or computed predictions instead of human scores.

The next group focused on testing a variety of models and parameters. **BRUMS** (Hettiarachchi and Ranasinghe, 2020) worked with ELMo, BERT, Flair (Akbik et al., 2018), Transformer-XL (Dai et al., 2019) and XLNet (Yang et al., 2019). Their final submission made use of stacked embeddings proposed by Akbik et al. (2018). They won the Finnish Subtask 2, ended second in the two Slovene ones and performed very well in the two English ones. The **Hitachi** team (Morishita et al., 2020) looked at BERT and XML-R. Their main insight was that for every language, the layers from the center to the end where always the best performing ones, however while BERT performed best in the last layer, XLM-R did in the center one, suggesting their inner structure is organised differently. They won the Slovene Subtask 1, finished second in the two Croatian subtasks and performed competitively in the English ones. To conclude with this group **JUSTMasters** (Al-Khdour et al., 2020) tested several models, parameters and their own strategy to combine models. They achieved very good performance, especially in the English Subtask 2. However, in order to optimise their system, they made many more submissions than allowed in the competition; we therefore leave them out of the official ranking.

With a more multilingual approach, **BabelEncoding** (Costella Pessutto et al., 2020) proposed a solution in which they translated the contexts and target words to many languages and then used a weighted combination of monolingual pretrained non contextualised embeddings and BERT embeddings. Their idea is that the translation not only brings new resources but the process itself can produce useful information, for example to disambiguate. The approach works very well for the less resourced languages, being clearly the best system in that category, in both Subtask 1 and 2. Their system won Subtask 1 and 2 for Croatian (by a healthy margin) and 2 for Slovene, ending third in the Slovene Subtask 1 and third and second in the two Finnish ones.

The **MultiSem** team (Soler and Apidianaki, 2020) collected 5 different datasets in order to fine-tune their BERT models, most of them automatically generated from previous datasets to increase contextual influence. As an example, ukWaC-subs was created by substituting target words by either: a correct substitute; a word that could be the right substitute in other circumstances but it is not in this context; or a random word. The datasets included WiC, which when used to fine tune the model resulted in the best performance for Subtask1, giving them a third place. The approach works very well, giving a very consistent performance in all categories, and significantly improving the non fine-tuned model from a ρ=0.715 and 0.661 per subtask, to a ρ=0.760 and 0.718 respectively.

Ferryman's focus (Chen et al., 2020) was clearly the English Subtask 1, which they won with a modification of BERT in which they fed the TF-IDF score of the words to the model, thus incorporating information about the general importance of words. The system does very well at predicting the change between contexts, but surprisingly poorly at predicting similarity itself, ending last in the English Subtask 2 and second from the last in Croatian and Slovene.

The starting point of **CitiusNLP** (Gamallo, 2020) was the idea that, even if BERT seems to be able to encode syntactic structure, it doesn't seem to make use of it. They created a linguistically motivated system that relied in dependency to create predictions. However, its performance was considerably worse than BERT's and their actual submissions are based on a standard BERT model.

Finally, the **Will_Go** team (Bao et al., 2020) looked at different ways to measure similarity between embeddings, mixing euclidean distance with the most common cosine similarity and several others not

described in their paper. The combination works well, they achieved a second place in the English Subtask 1 and won the Finnish Subtask 1.

8 Conclusion

We resented the SemEval-2020 Task on *Graded Word Similarity in Context* and introduced our new dataset *CoSimLex*. We provided the motivation behind their design choices and described the annotation process. The task received a good number of submissions and system description papers (15 and 11 respectively). We hope both the task and the dataset will be useful for researchers looking into how state-of-the-art systems capture context, and help promote the use of psychologically and cognitively inspired ideas in our field. Some of the interesting highlights were good performance of WordNet-based sense embeddings, the improvements achieved in less-resourced languages by simply translating the input, how the explicit feeding of an "old-fashioned" feature like TF-IDF improved a very modern system's performance, and the power of well designed, automatically created datasets for fine-tuning.

Additional and more detailed analyses of the dataset and task results will follow as part of future work. Areas to be investigated include the impact of different similarity ranges and degrees of polysemy, and more detailed qualitative analysis of the differences in annotation and between systems.

Acknowledgements

This research is supported by the 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 reflect only the authors' views and the Commission is not responsible for any use that may be made of the information it contains. Carlos S. Armendariz is also supported by the EPSRC and AHRC Centre for Doctoral Training in Media and Arts Technology (EP/L01632X/1); Matthew Purver is also supported by the EPSRC project Streamlining Social Decision Making for Improved Internet Standards (SoDe- Stream, EP/S033564/1). The work of Ivan Vulić is supported by the ERC Consolidator Grant LEXICAL (no. 648909).

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649.

Nour Al-Khdour, Mutaz Bni Younes, Malak Abdullah, and AL-Smadi Mohammad. 2020. JUSTMasters at SemEval-2020 Task 3: Multilingual deep learning model to predict the effect of context in word similarity. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Carlos S. Armendariz, Matthew Purver, Matej Ulčar, Senja Pollak, Nikola Ljubešić, and Mark Granroth-Wilding. 2020. CoSimLex: A resource for evaluating graded word similarity in context. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5878–5886, Marseille, France, May. European Language Resources Association.

Wei Bao, Hongshu Che, and Jiandong Zhang. 2020. Will_Go at SemEval-2020 Task 3: An accurate model for predicting the (graded) effect of context in word similarity based on BERT. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Jose Camacho-Collados, Mohammad Taher Pilehvar, Nigel Collier, and Roberto Navigli. 2017. SemEval-2017 Task 2: Multilingual and cross-lingual semantic word similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 15–26.

Weilong Chen, Xin Yuan, Sai Zhang, Jiehui Wu, Yanru Zhang, and Yang Wang. 2020. Ferryman at SemEval-2020 Task: BERT with TFIDF-weighting for predicting the effect of context in word similarity. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

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

Lucas Rafael Costella Pessutto, Viviane P. Moreira, Tiago de Melo, and Altigran da Silva. 2020. BabelEncoding at SemEval-2020 Task 3: Contextual similarity as a combination of multilingualism and language models. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

David A. Cruse. 1986. *Lexical semantics*. Cambridge university press.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy, July. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. 2013. Measuring word meaning in context. *Computational Linguistics*, 39(3):511–554.

Vyvyan Evans and Melanie Green. 2018. *Cognitive Linguistics: An Introduction*. Routledge.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. *ACM Transactions on information systems*, 20(1):116–131.

Pablo Gamallo. 2020. CitiusNLP at SemEval-2020 Task 3: Comparing two approaches for word vector contextualization. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Peter Gärdenfors. 2000. *Conceptual Spaces: The Geometry of Thought*. MIT Press.

Peter Gärdenfors. 2014. *The Geometry of Meaning: Semantics Based on Conceptual Spaces*. MIT Press.

Filip Ginter, Jan Hajič, Juhani Luotolahti, Milan Straka, and Daniel Zeman. 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.

Helena Gomez-Adorno, Gemma Bel-Enguix, Jorge Reyes-Magaña, Benjamin Moreno, Ramon Casillas, and Daniel Vargas. 2020. MineriaUNAM at SemEval-2020 Task 3: Predicting contextual word similarity using a centroid based approach and word embeddings. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Hansi Hettiarachchi and Tharindu Ranasinghe. 2020. BRUMS at SemEval-2020 Task 3: Contextualised embeddings for predicting the (graded) effect of context in word similarity. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

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

Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng. 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.

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

Daniel Loureiro and Jose Camacho-Collados. 2020. Don't neglect the obvious: On the role of unambiguous words in word sense disambiguation.

Daniel Loureiro and Alípio Jorge. 2019. Language modelling makes sense: Propagating representations through WordNet for full-coverage word sense disambiguation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5682–5691, Florence, Italy, July. Association for Computational Linguistics.

Somaia Mahmoud and Marwan Torki. 2020. AlexU-AUX-BERT at SemEval-2020 Task 3: Improving BERT contextual similarity using multiple auxiliary contexts. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

David E Meyer and Roger W Schvaneveldt. 1971. Facilitation in recognizing pairs of words: evidence of a dependence between retrieval operations. *Journal of experimental psychology*, 90(2):227.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

George A Miller. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Terufumi Morishita, Gaku Morio, Hiroaki Ozaki, and Toshinori Miyoshi. 2020. Hitachi at SemEval-2020 Task 3: Exploring the representation spaces of transformers for human sense word similarity. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017. Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the Association for Computational Linguistics*, 5:309–324.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 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.

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

Mohammad Taher Pilehvar and Jose Camacho-Collados. 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.

Aina Gari Soler and Marianna Apidianaki. 2020. MultiSem at SemEval-2020 Task 3: Fine-tuning BERT for lexical meaning. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Li Tang. 2020. UZH at SemEval-2020 Task 3: Combining BERT with WordNet sense embeddings to predict graded word similarity changes. In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Matej Ulčar and Marko Robnik-Šikonja. 2020. High quality ELMo embeddings for seven less-resourced languages. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4731–4738, Marseille, France, May. European Language Resources Association.

Viljami Venekoski and Jouko Vankka. 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 in Linköping Electronic Conference Proceedings, pages 231–236. Linköping University Electronic Press, Linköpings universitet.

Ivan Vulić, Simon Baker, Edoardo Maria Ponti, Ulla Petti, Ira Leviant, Kelly Wing, Olga Majewska, Eden Bar, Matt Malone, Thierry Poibeau, Roi Reichart, and Anna Korhonen. 2020. Multi-SimLex: A large-scale evaluation of multilingual and cross-lingual lexical semantic similarity. *arXiv preprint arXiv:2003.04866*.

Massimo Warglien and Peter Gärdenfors. 2015. Meaning negotiation. In *Applications of conceptual spaces*, pages 79–94. Springer.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

A Appendix: Survey Example

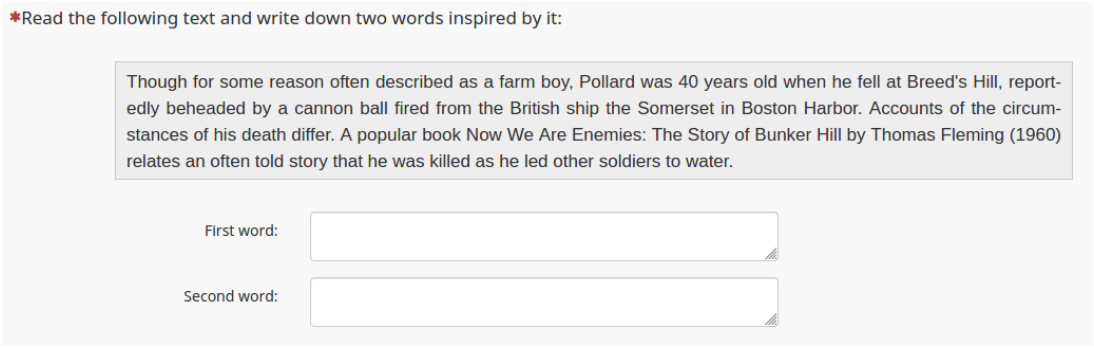


Figure 3: First page shown for each word pair annotation task: annotators must read the context and come up with two words inspired by it. At this point, the word pair to be scored is not known to the annotator.

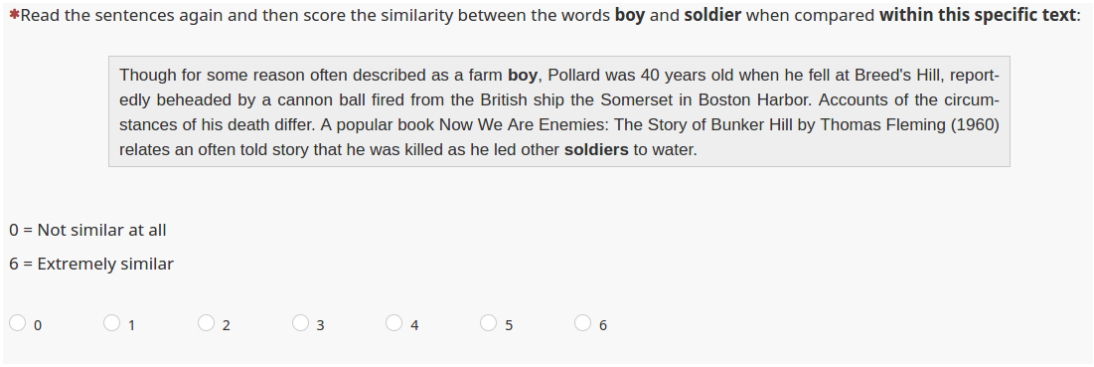


Figure 4: Second page shown for each word pair annotation task: the same context is now shown with the target words in bold, and annotators must give a similarity score for the word pair within that particular context.

B Appendix: Less-resourced Examples

B.1 Croatian

Word1: nov Word2: svjež	**SimLex (English)**: μ 6.83 σ 1.2
Context1	**Context1:** μ 9.49 σ 1.05
U jesen 1175. Fridrik je zamolio **svježe** trupe iz Njemačke. Prije svega Henrik Lav kao najmoćniji knez i vladar Bavarske odbio je caru poslati **nove** vojnike uvjetujući to prepuštanjem Goslara s bogatim rudnicima srebra.	
Context2	**Context2:** μ 1.85 σ 2.42
Proučavanje upalnih promjena dokazao je da ulaženje bijelih krvnih tjelešaca u tkivo uzrokuje gnojenje. Po njegovoj teoriji, rak nastaje iz emrionalnih stanica, razbacanih po organizmu. Uveo je **nove** metode istraživanja, npr. smrzavanje **svježeg** tkiva i pravljenje mirkoskopskih rezova.	
	P-Value: 2.4×10^{-5}

Figure 5: Example from the Croatian dataset, showing a word pair with two contexts, each with mean and standard deviation of human similarity judgements. The P-Value shown is the result of a Mann-Whitney U test.

B.2 Slovene

Word1: zgodba **Word2: tema**	**SimLex (English)**: μ 5 σ 1.7
Context1	**Context1:** μ 0.167 σ 0.527
V **zgodbi** Čajanka za psa mačka in papagaja, se cunjasta dvojčica Nina sooča s strahom. Ker je še majhna deklica se boji **teme**, toda na pomoč ji prihiti punčka in škratje Copatki, ki Nini predlagajo naj se poveselijo in priredijo čajanko. Skupaj s papagajem, psom in mačkom priredijo čajanko in pozabijo na strah.	
Context2	**Context2:** μ 6.3 σ 1.11
Koreografijo je sestavil Jamal Sims, ki je z Miley Cyrus sodeloval že pri plesu za pesem »Hoedown Throwdown«. Miley Cyrus in Jamal Sims sta skupaj sestavila koreografijo, ki bi se ujemala z **zgodbo** v pesmi, in nazadnje vse skupaj predstavila Robertu Halsu, ki si je »takoj zamislil, kako bo vse skupaj izgledalo«. V zvezi s **temo** videospota je Miley Cyrus povedala: »Mislim, da videospot razloži, da moje življenje ne izključuje življenj drugih ljudi.	
	P-Value: 5.1×10^{-5}

Figure 6: Example from the Slovene dataset, showing a word pair with two contexts, each with mean and standard deviation of human similarity judgements. The P-Value shown is the result of a Mann-Whitney U test.

B.3 Finnish

Word1: rikos **Word2: varkaus**	**SimLex (English)**: μ 7.53 σ 1.32
Context1	**Context1:** μ 4.33 σ 2.38
Valistuksen vaikutuksesta häpeärangaistuksista vähitellen luovuttiin. Esimodernissa Euroopassa häpeärangaistuksiin johtivat etupäässä pienehköt **rikokset**, kuten solvaukset ja häiritsevä juopumus, mutta myös esimerkiksi aviorikos ja **varkaus**. Häpeärangaistuksien toteuttamistavat vaihtelivat alueellisesti.	
Context2	**Context2:** μ 0 σ 0
Tekoja voidaan siis pitää pääosin laittomina, koska tuolloin ei ollut käytettävissä kuolemanrangaistuksen sallivaa, asianmukaista lainsäädäntöä. Sisällissodan jälkeen laaditulla armahduslailla vapautettiin myös valkoisen osapuolen edustajat vastuusta mahdollisesti tekemistään **rikoksista**, joten jonkinlainen ymmärrys teloitusten laittomuudesta oli ollut olemassa jo tuolloin. Kuolemantuomioiden langettamista jatkoi **Varkauden** kenttäoikeus, jonka lainmukaisuudesta voidaan olla myös hyvin erimielisiä.	
	P-Value: 3.3×10^{-5}

Figure 7: Example from the Finnish dataset, showing a word pair with two contexts, each with mean and standard deviation of human similarity judgements. The P-Value shown is the result of a Mann-Whitney U test. This is a very particular example, while "rikos" translates as "crime" and "varkaus" as "theft", there is a town named "Varkaus", which is the meaning of the word in the second context. This is the reason why all the annotators, accurately scored the similarity of the two words as 0 in the second context.

Appendix F: Incremental Composition in Distributional Semantics

Incremental Composition in Distributional Semantics*

Matthew Purver,[1,2] Mehrnoosh Sadrzadeh,[3] Ruth Kempson,[4]
Gijs Wijnholds,[1] Julian Hough[1]
[1]School of Electronic Engineering and Computer Science,
Queen Mary University of London
{m.purver,g.j.wijnholds,j.hough}@qmul.ac.uk
[2]Department of Knowledge Technologies, Jožef Stefan Institute
[3]Department of Computer Science, University College London
m.sadrzadeh@ucl.ac.uk
[4]Department of Philosophy, King's College London
ruth.kempson@kcl.ac.uk

Abstract

Despite the incremental nature of Dynamic Syntax (DS), the semantic grounding of it remains that of predicate logic, itself grounded in set theory, so is poorly suited to expressing the rampantly context-relative nature of word meaning, and related phenomena such as incremental judgements of similarity needed for the modelling of disambiguation. Here, we show how DS can be assigned a compositional distributional semantics which enables such judgements and makes it possible to incrementally disambiguate language constructs using vector space semantics. Building on a proposal in our previous work, we implement and evaluate our model on real data, showing that it outperforms a commonly used additive baseline. In conclusion, we argue that these results set the ground for an account of the non-determinism of lexical content, in which the nature of word meaning is its dependence on surrounding context for its construal.

1 Introduction

At the core of Dynamic Syntax (DS) as a grammar formalism has been the claim that the traditional concept of syntax — principles underpinning a set

*This is a post-peer-review, pre-copyedit version of an article published in the Journal of Logic, Language and Information. The final authenticated version will be available online at: http://dx.doi.org/TBD.

of structures inhabited by strings — should be replaced by a dynamic perspective in which syntax is a set of procedures for incrementally building up representations of content relative to context. Central to this claim has been the concept of underspecification and update, with partial content-representations being progressively built up on a word-by-word basis, allowing the emergence of progressively established content. Being a grammar formalism, DS underpins both speaker actions and hearer actions, with the immediate consequence of being able to characterise directly the to-and-fro dynamic of conversational dialogue. In informal conversations, people fluently switch between speaking and listening in virtue of each agent constructing incrementally evolving representations as driven by the words uttered and the procedures they induce. As a result, any one of them is able to adopt the lead role in this process at any stage. This was one of many confirmations of the general stance of incorporating within the grammar formalism a reflection of time incrementality (Kempson et al., 2016, *inter alia*).

Within this framework, words have been defined as inducing procedures for developing tree-theoretic representations of content (Cann et al., 2005; Kempson et al., 2001, 2011). However throughout much of the DS development there has been one major conservatism. The concept of semantic representation was taken, along broadly Fodorian lines, as involving a simple word-concept mapping. This was defined by Kempson et al. (2001) as a mapping onto expressions of the epsilon calculus, with its set-theoretically defined semantics (an epsilon term being defined as denoting a witness for the constructed arbitrary name manipulated in natural deduction systems of predicate calculus), a stance adopted as commensurate with the broadly proof-theoretic perspective of DS, and additionally motivated by the character of epsilon terms under development as displaying a growing reflection of the context within which they are constructed. Though attractive in matching the characteristic entity-typing of noun phrases, such a concept of word meaning is both too narrow in reflecting only what is expressible within predicate logic terms, and yet too strong in defining fixed extensions as content of the individual expressions, a move which provides no vehicle for addressing how content words display considerable context-dependence. In effect, the problem of explaining what meaning can be associated with a word as the systematic contribution it makes to sentence meaning without positing a veritable Pandora's box of ambiguities was not addressed. The same is true in many other frameworks: formal semanticists have by and large remained content with defining ambiguities whenever denotational considerations seemed to warrant them; and Partee (2018) cites the context-dependence of lexical semantics as a hurdle for which such a methodology does not appear to offer any natural means of addressing. And even within pragmatics, with its dedicated remit of explicating context-particular effects external to a standard competence model of grammar, and recent work on polysemy probing what this amounts to (Recanati, 2017; Carston, 2019), there nevertheless remains a tendency to invoke ambiguity involving discrete token-identical forms in the face of multiple interpretation potential, thereby leaving the phenomenon of natural language plasticity unexplained (Fretheim, 2019). For DS

as defined in (Kempson et al., 2001; Cann et al., 2005; Kempson et al., 2011), polysemy would thus also seem to remain a hurdle despite accounts of anaphora and ellipsis (see Kempson et al., 2015).

The challenge is this: words of natural language (NL) can have extraordinarily variable interpretations (even setting the problem of metaphor aside). A 'fire' in a grate is a warm welcome upon entering a house while a 'fire' in surrounding countryside causes widespread alarm. A 'burning' of a scone denotes a quite different process leading to quite different effects than 'burning' of a frying pan, or indeed 'burning' of a forest. The substance of the way in which such NL tokens are understood is deeply embedded within the contingent and culture-specific variability of perspectives which individual members of that community bring to bear in interaction with each other based on both supposedly shared knowledge of that language and their own practical and emotional experience. And such variation can occur when, within a single exchange, even a single speaker is able to shift construal for a single word, fragment by fragment as the participants finesse what they are talking. This is shown by the potential surface ungrammaticality of shared utterances (or *compound contributions*, Howes et al., 2011) which is in fact perfectly grammatical across speakers:

(1) A: I've almost completely burned the kitchen.
B: Did you burn..?
A: (interrupting) Myself? No, fortunately not. Well, only my hair

Yet, as long as the assumption that knowledge of language has to be modelled in some sense as prior to, hence independent of, any model of how that knowledge is put to use, this endemic context-relativity of even the basic units of language remains deeply intransigent; and the assumptions underpinning the long-held competence performance distinction have until very recently only been subject to minor modification amongst formal semanticists, despite the advocacy of need for more radical change from conversation analysts such as Schegloff (1984), psycholinguists such as Clark (1996) and Healey et al. (2018), and increasingly within cognitive neuroscience (e.g. Anderson, 2014).

Though DS purports to provide a general framework for modelling NL grammar in incremental terms, it was not until Purver et al. (2011) combined DS with Type Theory with Records (DS-TTR) that it became able to fully capture the incremental compositionality of semantic representation required to explain, for example, how people interactively co-construct shared utterances (see Purver et al., 2014). Even then, however, the challenge of modelling rampant lexical ambiguity was not addressed, and the attendant process of disambiguation also remained an open issue.

In previous work (Sadrzadeh et al., 2017, 2018b,a) we showed how in principle one can address these problems within the DS framework via the use of *distributional* or *vector space* semantics (VSS). By representing word meanings as vectors within a continuous space, VSS approaches can provide not only quantitative tools for measuring graded relations between words such as relatedness and similarities of meaning, but also a natural way to express the non-determinism of a word's construal from a denotational perspective, even relative

to context (see e.g. Coecke, 2019, for initial work on how such an approach can model the change of meaning through discourse). Moreover, we believe that the combination of a vector-space rendition of word meaning with the DS process-oriented characterisation of NL syntax is timely and of cross-disciplinary significance, as it promises to fill a niche within cognitive neuroscience where the emphasis is increasingly one of defining cognitive abilities in processual rather than representational terms – see discussion in Section 6.

In that earlier work, we outlined a theoretical approach to incorporating VSS within DS (Sadrzadeh et al., 2017, 2018b); we then demonstrated with toy examples how this approach might work to capture incremental measures of plausibility, and suggested that it might also be applied to word sense disambiguation (Sadrzadeh et al., 2018a). In this paper, we first review that approach (Sections 2 and 3), and then continue to explore this research program by extending that work: in Section 4 we show in detail how the proposed model can be applied to a word sense disambiguation task, and in Section 5 we implement the theoretical model using real data, and evaluate it on existing datasets for word sense disambiguation. Our approach addresses the polysemy problem directly by adopting the presumption that even relatively unorthodox cases of putative ambiguity such as the verbs *slump*, *tap*, and *dribble* can be analysed from a unitary processual base (these cases are where Vector Space Semantics, since its early days, has been known to apply most successfully; see e.g. the original work of Schütze, 1998). We take the corpus-based approach to word meaning with vector spaces deducible from possible containing contexts within large scale corpora as a formal analogue to the contingent and highly culture-specific variability of word meanings and usages. We provide evidence from the corpora on degrees of similarities between variations of finished and unfinished utterances, present accuracy results, and explore the effect of incrementality on an existing disambiguation dataset. In conclusion, we reflect on how VSS combined with DS assumptions opens up the possibility of modelling the general non-determinism of NL meaning in the light of this incremental interactive perspective with its shift away from direct pairings of string and denotational content to a more dynamic and non-deterministic stance.

2 Background

2.1 Dynamic Syntax and Incremental Semantic Parsing

Dynamic Syntax (DS) provides a strictly incremental formalism relating word sequences to semantic representations. Conventionally, these are seen as trees decorated with semantic formulae that are terms in a typed lambda calculus (Kempson et al., 2001, chapter 9):

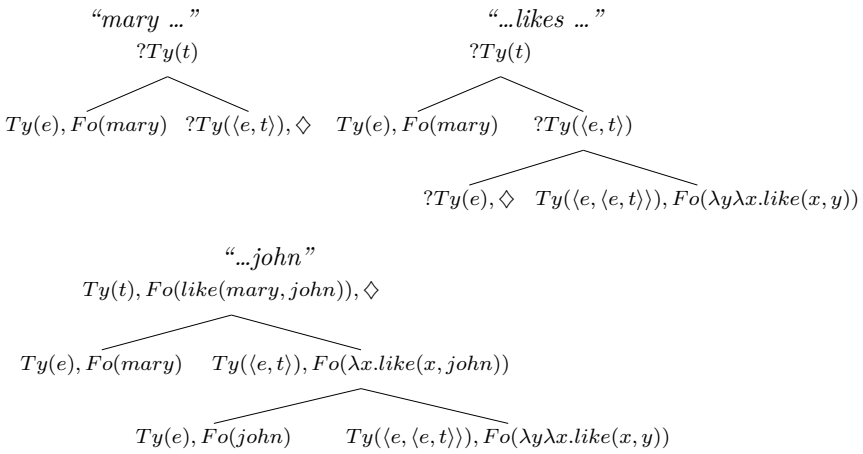
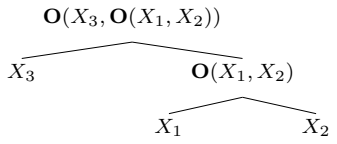


Figure 1: DS parsing as semantic tree development, for an utterance of the simple sentence *"Mary likes John"*.



"In this paper we will take the operation **O** *to be function application in a typed lambda calculus, and the objects of the parsing process […] will be terms in this calculus together with some labels; […]"*

This permits analyses of the semantic output of the word-by-word parsing process in terms of partial semantic trees, in which nodes are labelled with types Ty and semantic formulae Fo, or with requirements for future development (e.g. $?Ty$. $?Fo$), and with a pointer $\Diamond$ indicating the node currently under development. This is shown in Figure 1 for the simple sentence *Mary likes John*. Phenomena such as conjunction, apposition and relative clauses are analysed via LINKed trees (corresponding to semantic conjunction). For reasons of space we do not present an original DS tree for these here (see section 2.5 of the introduction to this volume); an example of a non-restrictive relative clause linked tree labelled with vectors is presented in Figure 4.

The property of strict word-by-word incrementality inherent in all versions of DS makes it a good candidate for modelling language in natural human interaction. Speakers and hearers in dialogue can swap roles during sentences, without holding to notions of traditional syntactic or semantic constituency (see Howes et al. (2011) and example (1)). Speakers often produce incomplete output, and hearers manage to understand the meaning conveyed so far. In order to perform these ordinary feats, a suitable parsing and generation model must deal in incremental representations which capture the semantic content built at any point, and reflect grammatical constraints appropriately, and this is something DS does well (Cann et al., 2007). Accordingly, DS analyses of many dialogue phenomena have been produced: for example, shared utterances (Purver et al., 2014), self-repair (Hough and Purver, 2012), and backchannelling (Eshghi et al., 2015).

Much recent work in dialogue understanding takes a purely machine-learning

approach, learning how to encode input utterances into representations which can be decoded into appropriate follow-ups, without requiring prior knowledge of dialogue phenomena or structure (see e.g. Vinyals and Le, 2015). However, while these models can show good accuracy in terms of understanding speaker intentions and generating suitable output, their representations are suitable only for the task and domain for which they are learned, and do not learn meaningful information about important linguistic phenomena like self-repair (Hupkes et al., 2018). Structured grammar-based approaches like DS can therefore contribute more general, informative models, from which robust versions can be learned (Eshghi et al., 2017).

2.2 DS and Semantic Representation

As presented above, however, and in its original form, DS assumes semantic formulae expressed in a standard symbolic predicate logic, and therefore not well suited to the problems of non-determinism, (dis)similarity and shift in word meanings discussed in Section 1. But the DS formalism is in fact considerably more general. To continue the quotation above:

> *"[…] it is important to keep in mind that the choice of the actual representation language is not central to the parsing model developed here. […] For instance, we may take* X_1, X_2, X_3 *to be feature structures and the operation* **O** *to be unification, or* X_1, X_2, X_3 *to be lambda terms and* **O** *Application, or* X_1, X_2, X_3 *to be labelled categorial expressions and* **O** *Application: Modus Ponens, or* X_1, X_2, X_3 *to be DRSs and* **O** *Merging."*

This generality has been exploited in more recent work: Purver et al. (2010, 2011) outlined a version in which the formulae are *record types* in Type Theory with Records (TTR, Cooper, 2005) in DS-TTR; and Hough and Purver (2012) show how this can confer an extra advantage – the incremental decoration of the *root* node, even for partial trees, with a maximally specific formula via type inference, using the TTR merge operation $\wedge$ as the composition function. In the latter account, underspecified record types decorate requirement nodes, containing a type judgement with the relevant type (e.g. $[x : e]$ at type $?Ty(e)$ nodes)– see Fig. 2 for a DS-TTR parse of *"Mary likes John"*. Hough and Purver (2017) show that this underspecification can be given a precise semantics through record type lattices: the dual operation of merge, the minimum common super type (or join) $\vee$ is required to define a (probabilistic) distributive record type lattice bound by $\wedge$ and $\vee$. The interpretation process, including reference resolution, then takes the incrementally built top-level formula and checks it against a type system (corresponding to a world model) defined by a record type lattice. Implicitly, the record type on each node in a DS-TTR tree can be seen to correspond to a potential set of type judgements as sub-lattices of this lattice, with the appropriate underspecified record type (e.g. $[x : e]$) as their top element, with a probability value for each element in the probabilistic

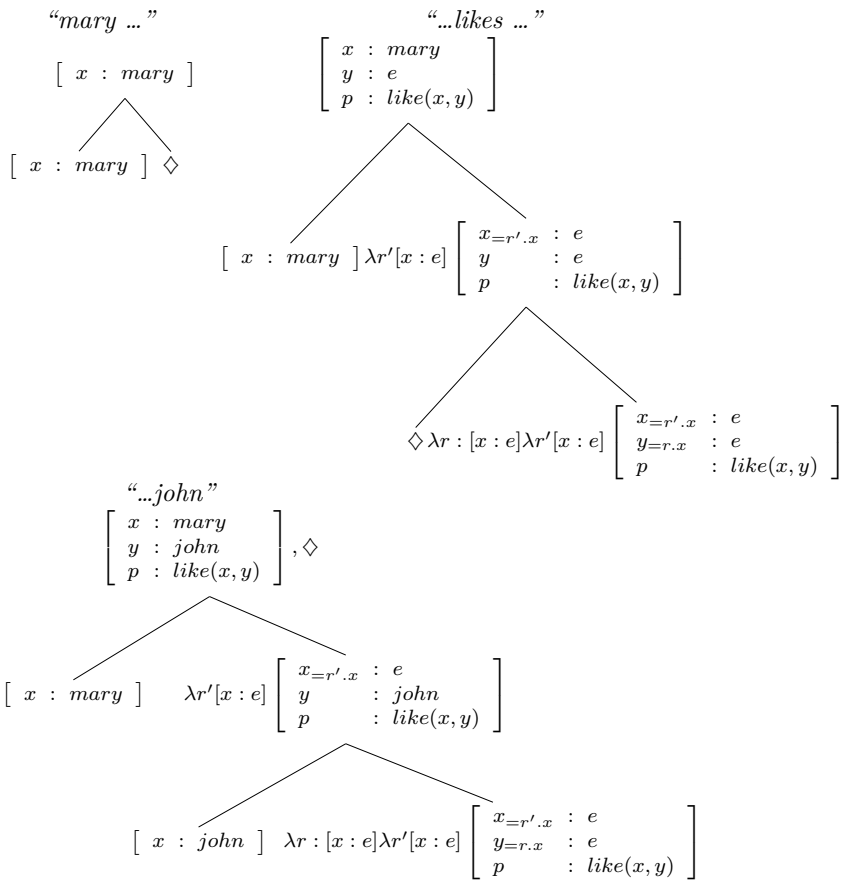


Figure 2: DS-TTR parse of *"Mary likes John"*

TTR version. Building on this, Sadrzadeh et al. (2018b) took the first steps in showing how equivalent underspecification, and narrowing down of meaning over time can be defined for vector space representations with analogous operations to $\mathbb{A}$ and $\mathbb{V}$ — this gives the additional advantages inherent in vector space models such as established techniques for computing similarity judgements between word, phrase and sentence representations.

3 Compositional Vector Space Semantics for DS

Vector space semantics are commonly instantiated via lexical co-occurrence, based on the *distributional hypothesis* that meanings of words are represented by the distributions of the words around them; this is often described by Firth's claim that 'you shall know a word by the company it keeps' (Firth, 1957). More specifically, the methodology of distributional semantics has involved taking

very large corpus collections as the data source and defining the content of a word as a function of the number of times it occurs in relation to other relevant expressions in that collection, as determined by factors such as similarity and dependency relations with such expressions. This can be implemented by creating a co-occurrence matrix (Rubenstein and Goodenough, 1965), in which the columns are labelled by context words and the rows by target words; the entry of the matrix at the intersection of a context word c and a target word t is a function (such as TF-IDF or PPMI) of the number of times t occurred in the context of c (as defined via e.g. a lexical neighbourhood window, a dependency relation, etc.). The meaning of each target word is represented by its corresponding row of the matrix. These rows are embedded in a vector space, where the distances between the vectors represent degrees of semantic similarity between words (Schütze, 1998; Lin, 1998; Curran, 2004). Alternatively, rather than instantiating these vectors directly from co-occurrence statistics, the vectors can be learned (usually via a neural network) in order to predict co-occurrence observations and thus encode meaning in a similar way (see e.g. Baroni et al., 2014b, for a comparison of these methods).

Distributional semantics has been extended from word level to sentence level, where compositional operations act on the vectors of the words to produce a vector for the sentence. Existing models vary from using simple additive and multiplicative compositional operations (Mitchell and Lapata, 2010) to operators based on fully fledged categorial grammar derivations, e.g. pregroup grammars (Coecke et al., 2010; Clark, 2013), the Lambek Calculus (Coecke et al., 2013), Combinatory Categorial Grammar (CCG) (Krishnamurthy and Mitchell, 2013; Baroni et al., 2014a; Maillard et al., 2014) and related formalisms, such as multimodal Lambek Calculi (Moortgat and Wijnholds, 2017). However, most work done on distributional semantics has not been directly compatible with incremental processing, although first steps were taken in Sadrzadeh et al. (2017) to develop such an incremental semantics, using a framework based on a categorial grammar as opposed to in the DS formalism, i.e. one in which a full categorial analysis of the phrase/sentence was the obligatory starting point.

Compositional vector space semantic models have a complementary property to DS. Whereas DS is agnostic to its choice of semantics, compositional vector space models are agnostic to the choice of the syntactic system. Coecke et al. (2010) show how they provide semantics for sentences based on the grammatical structures given by Lambek's pregroup grammars (Lambek, 1997); Coecke et al. (2013) show how this semantics also works starting from the parse trees of Lambek's Syntactic Calculus (Lambek, 1958); Wijnholds (2017) shows how the same semantics can be extended to the Lambek-Grishin Calculus; and (Krishnamurthy and Mitchell, 2013; Baroni et al., 2014a; Maillard et al., 2014) show how it works for CCG trees. These semantic models homomorphically map the concatenation and slashes of categorial grammars to tensors and their evaluation/application/composition operations, as shown by (Maillard et al., 2014), all of which can be reduced to tensor contraction.

In DS terms, structures X_1, X_2, X_3 are mapped to general higher order ten-

sors, e.g. as follows:

$$\begin{array}{lcll} X_1 & \mapsto & T_{i_1 i_2 \cdots i_n} & \in \; V_1 \otimes V_2 \otimes \cdots V_n \\ X_2 & \mapsto & T_{i_n i_{n+1} \cdots i_{n+k}} & \in \; V_n \otimes V_{n+1} \otimes \cdots V_{n+k} \\ X_3 & \mapsto & T_{i_{n+k} i_{n+k+1} \cdots i_{n+k+m}} & \in \; V_{n+k} \otimes V_{n+k+1} \otimes \cdots V_{n+k+m} \end{array}$$

Each $T_{i_1 i_2 \cdots i_n}$ abbreviates the linear expansion of a tensor, which is normally written as follows:

$$T_{i_1 i_2 \cdots i_n} \equiv \sum_{i_1 i_2 \cdots i_n} C_{i_1 i_2 \cdots i_n} e_1 \otimes e_2 \otimes \cdots \otimes e_n$$

for e_i a basis of V_i and $C_{i_1 i_2 \cdots i_n}$ its corresponding scalar value. The **O** operations are mapped to contractions between these tensors, formed as follows:

$$\begin{array}{rcl} \mathbf{O}(X_1, X_2) & \mapsto & T_{i_1 i_2 \cdots i_n} T_{i_n i_{n+1} \cdots i_{n+k}} \\ & \in & V_1 \otimes V_2 \otimes \cdots \otimes V_{n-1} \otimes V_{n+1} \otimes \cdots \otimes V_{n+k} \\ \mathbf{O}(X_3, \mathbf{O}(X_1, X_2)) & \mapsto & T_{i_1 i_2 \cdots i_n} T_{i_n i_{n+1} \cdots i_{n+k}} T_{i_{n+k} i_{n+k+1} \cdots i_{n+k+m}} \\ & \in & V_1 \otimes V_2 \otimes \cdots \otimes V_{n-1} \otimes V_{n+1} \otimes \cdots \\ & & \cdots \otimes V_{n+k-1} \otimes V_{n+k+1} \otimes \cdots \otimes V_{n+k+m} \end{array}$$

In their most general form presented above, these formulae are large and the index notation becomes difficult to read. In special cases, however, it is often enough to work with spaces of rank around 3. For instance, the application of a transitive verb to its object is mapped to the following contraction:

$$T_{i_1 i_2 i_3} T_{i_3} = (\sum_{i_1 i_2 i_3} C_{i_1 i_2 i_3} e_1 \otimes e_2 \otimes e_3)(\sum_{i_3} C_{i_3} e_3) = \sum_{i_1 i_2} C_{i_1 i_2 i_3} C_{i_3} e_1 \otimes e_2$$

This is the contraction between a cube $T_{i_1 i_2 i_3}$ in $X_1 \otimes X_2 \otimes X_3$ and a vector T_{i_3} in X_3, resulting in a matrix in $T_{i_1 i_2}$ in $X_1 \otimes X_2$.

We take the DS propositional type $Ty(t)$ to correspond to a sentence space S, and the entity type $Ty(e)$ to a word space W. Given vectors T_i^{mary}, T_k^{john} in W and the (cube) tensor T_{ijk}^{like} in $W \otimes S \otimes W$, the tensor semantic trees of the DS parsing process of "*Mary likes John*" become as in Fig. 3.[1]

A very similar procedure is applicable to the linked structures, where conjunction can be interpreted by the μ map of a Frobenius algebra over a vector space, e.g. as in (Kartsaklis, 2015), or as composition of the interpretations of its two conjuncts, as in (Muskens and Sadrzadeh, 2016). The μ map has also been used to model relative clauses (Clark et al., 2013; Sadrzadeh et al., 2013, 2014). It *combines* the information of the two vector spaces into one. Figure 2 shows how it combines the information of two contracted tensors $T_i^{mary} T_{ij}^{sleep}$ and $T_i^{mary} T_{ij}^{snore}$.

DS *requirements* can now be treated as requirements for tensors of a particular order (e.g. $?W$, $?W \otimes S$ as above). If we can give these suitable vector-space

[1] There has been much discussion about whether sentence and word spaces should be the same or separate. In previous work, we have worked with both cases, i.e. when $W \neq S$ and when $W = S$.

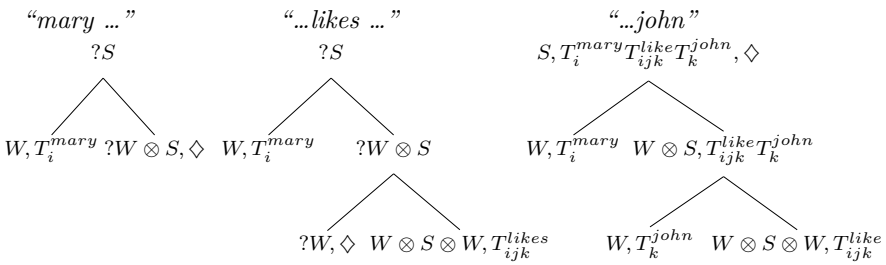


Figure 3: A DS with Vector Space Semantics parse of *"Mary likes John"*.

representations, we can then provide a procedure analogous to that of Hough and Purver (2012)'s incremental type inference procedure, allowing us to compile a partial tree to specify its overall semantic representation (at its root node). One alternative would be to interpret them as picking out an element which is *neutral* with regards to composition: the unit vector/tensor of the space they annotate. A more informative alternative would be to interpret them as enu-

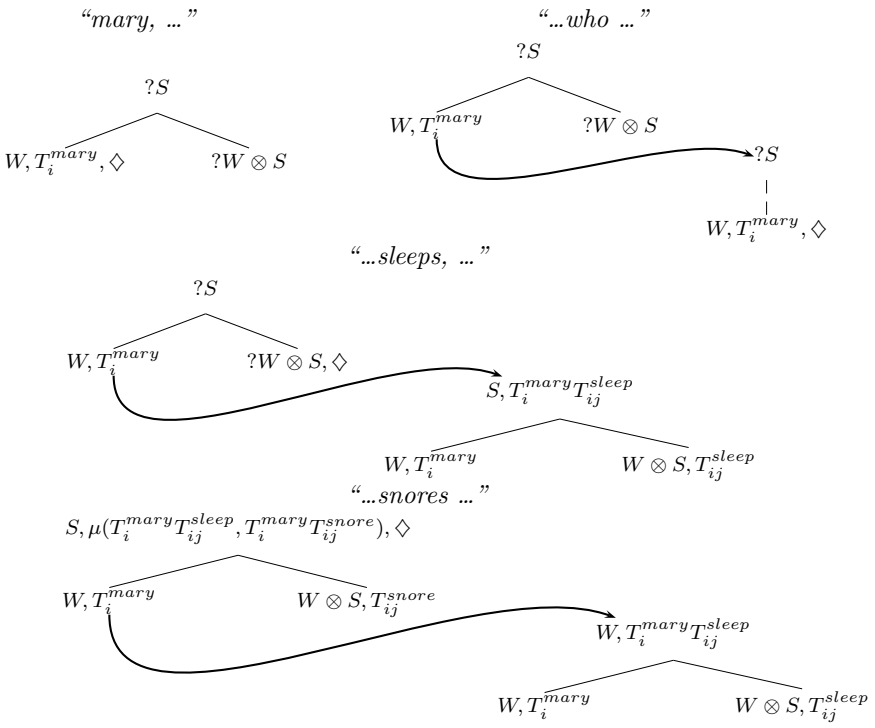


Figure 4: A DS with Vector Space Semantics parse of *"Mary, who sleeps, snores''*.

merating all the possibilities for further development. This can be derived from all the word vectors and phrase tensors of the space under question — i.e. all the words and phrases whose vectors and tensors live in W and in $W \otimes S$ in this case — by taking either the *sum* T^{+} or the *direct sum* $T^{\oplus}$ of these vectors/tensors. Summing will give us one vector/tensor, accumulating the information encoded in the vectors/tensors of each word/phrase; direct summing will give us a tuple, keeping this information separate from each other. This gives us the equivalent of a sub-lattice of the record type lattices described in (Hough and Purver, 2017), with the appropriate underspecified record type as the top element, and the attendant advantages for incremental probabilistic interpretation.

These alternatives all provide the desired compositionality, but differ in the semantic information they contribute. The use of the identity provides no extra semantic information beyond that contributed by the words so far; the sum gives information about the "average" vector/tensor expected on the basis of what is known about the language and its use in context (encoded in the vector space model); the direct sum enumerates/lists the possibilities. In each case, more semantic information can arrive later as more words are parsed. The best alternative will depend on task and implementation. In the experiments below, we implement and compare all these three methods.

4 Incremental Disambiguation

In this section, we show how our model can be applied to a common task in compositional distributional semantics: disambiguation of verb meanings.

4.1 A Disambiguation Task

Verbs can have more than one meaning and their contexts, e.g. their subjects, objects and other elements, can help disambiguate them. In compositional distributional semantics, this has been modelled by comparing different hypothesized paraphrases for a sentence, one for each of the meanings of the verb, and then measuring the degree of semantic similarity between the vectors for these hypothesized paraphrased sentences and the original sentence (the one containing the ambiguous verb). The sentence that is closer to the original sentence will then be returned as the one containing the disambiguated meaning of the verb. For instance, consider the verb *slump*; it can mean 'slouch' in the context of an utterance with *"shoulders"* as its subject, or it can mean 'decline' in the context of an utterance with *"sales"* as its subject. This procedure is implemented in compositional distributional semantics by building vectors for the following sentences:

"Shoulders slumped", *"Shoulders slouched"*, *"Shoulders declined"*.
"Sales slumped", *'Sales slouched"*, *"Sales declined"*

The semantic distances, e.g. the cosine distance, between these vectors are employed to see which ones of these sentences are closer to each other. If "x

slumped" is closest to "x slouched", then it is concluded that an utterance of *"slump"* means 'slouch' in the context of "x". This idea was used by Mitchell and Lapata (2010) to disambiguate intransitive verbs using their subjects as context. They showed that the compositional distributional methods work better than simple distributional methods: comparing distances between composed sentence representations gives more accurate paraphrase disambiguation than simply comparing the vectors of the individual verbs.

To test this, they used a dataset of sentences arranged in pairs:

Sentence1	Sentence2	Landmark
shoulders slumped	*shoulders declined*	LOW
shoulders slumped	*shoulders slouched*	HIGH
sales slumped	*sales declined*	HIGH
sales slumped	*sales slouched*	LOW

Each entry of the dataset consists of a pair of sentences and a similarity landmark (LOW, HIGH). Each sentence in the pair is created by replacing the verb of the first sentence with each of its two most orthogonal meanings. The meanings and the degrees of their orthogonality are drawn from WordNet and the synsets of the original verbs.

This dataset has been extended to transitive verbs, first by Grefenstette and Sadrzadeh (2011), using a set of frequent verbs from the British National Corpus (BNC, Burnard, 2000) and two of their meanings which are furthest apart using WordNet distances; and then by Kartsaklis and Sadrzadeh (2013) using a set of genuinely ambiguous verbs and their two eminent meanings introduced in (Pickering and Frisson, 2001) using eye tracking. Examples of the verbs of these are as follows:

Sentence1	Sentence2	Landmark
fingers tap table	*fingers knock table*	HIGH
fingers tap table	*fingers intercept table*	LOW
police tap telephone	*police knock telephone*	LOW
police tap telephone	*police intercept telephone*	HIGH
babies dribble milk	*babies drip milk*	HIGH
babies dribble milk	*babies control milk*	LOW
footballers dribble ball	*footballers control ball*	HIGH
footballers dribble ball	*footballers drip ball*	LOW

In compositional distributional semantics, one can build vectors for the words of these sentences and add or pointwise multiply them to obtain a vector for the whole sentence (Mitchell and Lapata, 2008). Alternatively, one can build vectors for nouns and tensors for adjectives and verbs (and all other words with functional types) and use tensor contraction to build a vector for the sentence (Grefenstette and Sadrzadeh, 2015; Kartsaklis and Sadrzadeh, 2013). It has been shown that some of the tensor-based models improve on the results of the additive model, when considering the whole sentence (Grefenstette and Sadrzadeh, 2015; Kartsaklis and Sadrzadeh, 2013; Wijnholds and Sadrzadeh, 2019); here, we focus on incremental composition as described above to investigate how the disambiguation process works word-by-word.

In the intransitive sentence datasets (Mitchell and Lapata, 2008), the disambiguation context only consists of the subject and verb, and the incremental process is fairly trivial (the ambiguity is only introduced when the verb is processed, and at that point the sentence is complete). We use intransitive examples to explain the principle first, but thereafter work with the transitive sentence datasets and their different variants.

4.2 An Incremental Disambiguation Procedure

In a nutshell, the disambiguation procedure is as follows: when we hear the word *"shoulders"* uttered, we can build a vectorial interpretation for the as-yet incomplete utterance, using the compositional distributional semantics of Dynamic Syntax as explained in Section 3 (and using either neutral identity information, or (direct) sum information about all the intransitive verbs and verb phrases that can follow). After we hear the verb *"slump"*, our uttered sentence is complete and we form a vector for it, again by using the compositional distributional semantics of DS (or the more traditional methods; the two should result in the same semantics for complete utterances). We can check the incremental behaviour of this process by one or more of the following steps:

1. The semantic vector of the unfinished utterance *"shoulders $\cdots$"* should be closer to the semantic vector of the sentence with the correct meaning of *"slump"* (i.e. to *"Shoulders slouched"*) than to the vector of the sentence with the incorrect meaning of *"slump"* (i.e. to *"Shoulders declined"*). Formally, using the cosine similarity measure of distributional semantics, the following should be the case:

$$\cos(\overrightarrow{shoulders\cdots}, \overrightarrow{shoulders\ slouched}) \geq \cos(\overrightarrow{shoulders\cdots}, \overrightarrow{shoulders\ declined})$$

Of course, the complete utterance *"shoulders slumped"* should also be closer to *"shoulders slouched"*. This is not incremental and has been verified in previous work (Mitchell and Lapata, 2008). We do not experiment with this case here, although, we might also expect, and could check, that it is closer to the full correct paraphrase than is the partial sentence:

$$\cos(\overrightarrow{shoulders\ slumped}, \overrightarrow{shoulders\ slouched}) \geq \cos(\overrightarrow{shoulders\cdots}, \overrightarrow{shoulders\ slouched})$$

2. Conversely, for an example in which the other verb paraphrase is appropriate: the semantic vector of the unfinished utterance *sales* $\cdots$ should be closer to the vector of the sentence *sales declined* than to that for *sales slouched*, and a full sentence be closer than an incomplete one:

$$\cos(\overrightarrow{sales\cdots}, \overrightarrow{sales\ declined}) \geq \cos(\overrightarrow{sales\cdots}, \overrightarrow{sales\ slouched})$$

$$\cos(\overrightarrow{sales\ slumped}, \overrightarrow{sales\ declined}) \geq \cos(\overrightarrow{sales\cdots}, \overrightarrow{sales\ declined})$$

3. We can also compare between the examples: the semantic vector of the unfinished utterance *shoulders* $\cdots$ should also be closer to the vector of the full sentence *shoulders slouched* than the vector of the unfinished utterance *"sales* $\cdots$ *"* is to that of the complete sentence *"sales slouched"*:

$$\cos(\overrightarrow{shoulders\cdots}, \overrightarrow{shoulders\ slouched}) \geq \cos(\overrightarrow{sales\cdots}, \overrightarrow{sales\ slouched})$$

And the other way around should also hold, that is, the vector of the unfinished utterance *sales* $\cdots$ should be closer to the vector of the uttered sentence *sales declined* than the vector of *shoulders* $\cdots$ is to *shoulders declined*.

$$\cos(\overrightarrow{sales\cdots}, \overrightarrow{sales\ declined}) \geq \cos(\overrightarrow{shoulders\cdots}, \overrightarrow{shoulders\ declined})$$

A symbolic generalisation of the above procedure for the *Sbj Vrb Obj* cases, which is the case we will experiment with, is presented below. In Section 5, we then provide evidence from real data, first giving a worked example for each of these cases, and then a large scale experimental evaluation.

Consider a verb *Vrb* that is ambiguous between two meanings *Vrb1* and *Vrb2*; suppose further that a subject *Sbj* makes more sense with the first meaning of the verb, that is with *Vrb1*, rather than with its second meaning, that is with *Vrb2*. This is because *Sbj* has more associations with *Vrb1*, e.g. since it has occurred more with *Vrb1* (or with verbs with similar tensors to *Vrb1*) than with *Vrb2* in a corpus. These correlations are interpreted in our setting as follows:

$$\cos(\overrightarrow{Sbj\cdots}, \overrightarrow{Sbj\ Vrb1\cdots}) \geq \cos(\overrightarrow{Sbj\cdots}, \overrightarrow{Sbj\ Vrb2\cdots})$$

We can extend this when we incrementally proceed and parse the verb *Vrb*. Now we can check the following:

$$\cos(\overrightarrow{Sbj\ Vrb\cdots}, \overrightarrow{Sbj\ Vrb1\cdots}) \geq \cos(\overrightarrow{Sbj\ Vrb\cdots}, \overrightarrow{Sbj\ Vrb2\cdots})$$

Here, we are incrementally disambiguating the unfinished utterance *Sbj Vrb* using the vector semantics of its subject *Sbj*, the tensor meaning of its verb *Vrb*, and the contraction (read composition) of the two. As we add more context and finish the incremental parsing of the utterances, similar regularities to the above are observed and we expect the corresponding degrees of semantic similarity to become more sharply distinguished as the object meaning *Obj* is added:

$$\cos(\overrightarrow{Sbj\cdots}, \overrightarrow{Sbj\ Vrb1\ Obj}) \geq \cos(\overrightarrow{Sbj\cdots}, \overrightarrow{Sbj\ Vrb2\ Obj})$$
$$\cos(\overrightarrow{Sbj\ Vrb\cdots}, \overrightarrow{Sbj\ Vrb1\ Obj}) \geq \cos(\overrightarrow{Sbj\ Vrb\cdots}, \overrightarrow{Sbj\ Vrb2\ Obj})$$
$$\cos(\overrightarrow{Sbj\ Vrb\ Obj}, \overrightarrow{Sbj\ Vrb1\ Obj}) \geq \cos(\overrightarrow{Sbj\ Vrb\ Obj}, \overrightarrow{Sbj\ Vrb2\ Obj})$$

The fronted object cases, *Obj Sbj Vrb*, such as in the sentence *The milk the baby dribbled* can also be dealt with, but are left to future work.

5 Evidence from Real Data

Of course, the real test is whether similarities calculated this way reflect those we would intuitively expect. In this section, we test this with some selected example sentences, using vectors and tensors calculated from real corpus data.

Our noun vectors are produced using `word2vec`, a commonly used neural network model for learning word vector representations (Mikolov et al., 2013): we use 300-dimensional vectors learned from the Google News corpus.[2] Our verb tensors are derived using the method of Grefenstette and Sadrzadeh (2011): the tensor $\overrightarrow{V}$ is the sum of $\overrightarrow{S} \otimes \overrightarrow{O}$ over the subject noun vectors $\overrightarrow{S}$ and object noun vectors $\overrightarrow{O}$ observed to co-occur with the verb in question in a large parsed corpus. Here we take the verb-subject/verb-object occurrences from the dependency-parsed version of UKWaC (Baroni et al., 2009), and use the same `word2vec` noun vectors; our verb tensors are therefore 300x300-dimensional matrices. To compose a sentence representation $\overrightarrow{A}$, we again follow Grefenstette and Sadrzadeh (2011), using point-wise multiplication of the verb tensor with the Kronecker product of the subject and object vectors (other methods are possible, and we explore these in the next section):

$$\overrightarrow{A} = \overrightarrow{V} \odot (\overrightarrow{S} \otimes \overrightarrow{O})$$

We start with an example from the dataset of Kartsaklis et al. (2013b): the ambiguous verb *dribble* has a different sense in the sentence *Footballers dribble balls* than in the sentence *Babies dribble milk*. If we take these senses to be roughly paraphrased as 'control' and 'drip', respectively, we can examine not only whether the full sentence representations are more similar to the appropriate paraphrases (as in the experiments of Kartsaklis et al., 2013b), but also whether this disambiguation is exhibited incrementally. Here, we take the option described above of representing unsatisfied requirements with the identity tensor I; we express similarities using the cosine similarity measure:

$$\text{similarity} = \cos(\overrightarrow{A}, \overrightarrow{B}) = \frac{\overrightarrow{A} \cdot \overrightarrow{B}}{\|\overrightarrow{A}\|\|\overrightarrow{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

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

First, we note that the expected pattern is observable between completed utterances (as expected, given the results of Mitchell and Lapata (2008) and Kartsaklis et al. (2013b)), with the representation for the complete sentence being more similar to the correct paraphrase (following Kartsaklis et al. (2013b) we simplify here by ignoring inflections such as plural suffixes and use the vectors and tensors for noun and verb root forms):

$$\begin{aligned} \cos(\overrightarrow{\textit{footballer dribble ball}}, \overrightarrow{\textit{footballer control ball}}) &= 0.3664 \\ \cos(\overrightarrow{\textit{footballer dribble ball}}, \overrightarrow{\textit{footballer drip ball}}) &= 0.2260 \end{aligned}$$

We can check the incremental behaviour by calculating and comparing similarities at incremental stages. First, after parsing only the subject, we see that *"Footballers $\cdots$ "* has a closer semantic similarity with *"Footballers control $\cdots$ "* than with *"Footballers drip $\cdots$ "*: that as you add to your unfinished utterances, its semantics builds up in a coherent way:

$$\begin{aligned} \cos(\overrightarrow{\textit{footballer}\cdots}, \overrightarrow{\textit{footballer control}\cdots}) &= 0.0860 \\ \cos(\overrightarrow{\textit{footballer}\cdots}, \overrightarrow{\textit{footballer drip}\cdots}) &= 0.0498 \end{aligned}$$

Next, after parsing the subject and verb, we again see the expected effect:

$$\begin{aligned} \cos(\overrightarrow{\textit{footballer dribble}\cdots}, \overrightarrow{\textit{footballer control}\cdots}) &= 0.3392 \\ \cos(\overrightarrow{\textit{footballer dribble}\cdots}, \overrightarrow{\textit{footballer drip}\cdots}) &= 0.2407 \end{aligned}$$

Similarly we can examine similarities with possible complete utterances, giving us a notion of incremental *expectation* in parsing; again we see an effect in the expected direction – the unfinished utterance *"Footballers $\cdots$ "* is semantically closer to *"Footballers dribble balls"* than to *"Footballers dribble milk"*, which is of course what semantically makes sense:

$$\begin{aligned} \cos(\overrightarrow{\textit{footballer}\cdots}, \overrightarrow{\textit{footballer dribble ball}}) &= 0.0046 \\ \cos(\overrightarrow{\textit{footballer}\cdots}, \overrightarrow{\textit{footballer dribble milk}}) &= 0.0019 \end{aligned}$$

And this also holds when the verb is parsed, i.e. as we carry on finishing the utterance, we get higher more reasonable similarity degrees:

$$\begin{aligned} \cos(\overrightarrow{\textit{footballer dribble}\cdots}, \overrightarrow{\textit{footballer dribble ball}}) &= 0.2246 \\ \cos(\overrightarrow{\textit{footballer dribble}\cdots}, \overrightarrow{\textit{footballer dribble milk}}) &= 0.0239 \end{aligned}$$

Similarly, for the unfinished utterance *"Babies $\cdots$ "* we obtain the following desirable results that agree with semantic incrementality:

$$\begin{aligned}
\cos(\overrightarrow{baby\ dribble\cdots}, \overrightarrow{baby\ drip\cdots}) &= 0.3269 \\
\cos(\overrightarrow{baby\ dribble\cdots}, \overrightarrow{baby\ control\cdots}) &= 0.3239 \\
\cos(\overrightarrow{baby\ dribble\ milk}, \overrightarrow{baby\ drip\ milk}) &= 0.3468 \\
\cos(\overrightarrow{baby\ dribble\ milk}, \overrightarrow{baby\ control\ milk}) &= 0.3291
\end{aligned}$$

However, this is not always the case; for the same utterance, the similarities calculated after parsing only the subject point in the opposite direction to that expected:

$$\begin{aligned}
\cos(\overrightarrow{baby\cdots}, \overrightarrow{baby\ drip\cdots}) &= 0.0573 \\
\cos(\overrightarrow{baby\cdots}, \overrightarrow{baby\ control\cdots}) &= 0.0932
\end{aligned}$$

It seems, therefore, that it must be the verb *dribble* and then even more strongly, the combination with the object *milk* that provides much of the disambiguating information in this case – perhaps babies alone are no more likely to drip than to control.

We have a similar situation for the ambiguous verb *tap*, its two meanings 'knock' and 'intercept', and the subject *"finger"* which disambiguates *"tap"* to its 'knock' meaning:

$$\begin{aligned}
\cos(\overrightarrow{finger\cdots}, \overrightarrow{finger\ knock\cdots}) &= 0.0667 \\
\cos(\overrightarrow{finger\cdots}, \overrightarrow{finger\ intercept\cdots}) &= 0.0534 \\
\cos(\overrightarrow{finger\ tap\cdots}, \overrightarrow{finger\ knock\cdots}) &= 0.6751 \\
\cos(\overrightarrow{finger\ tap\cdots}, \overrightarrow{finger\ intercept\cdots}) &= 0.4320 \\
\cos(\overrightarrow{finger\ tap\ wood}, \overrightarrow{finger\ knock\ wood} &= 0.7154 \\
\cos(\overrightarrow{finger\ tap\ wood}, \overrightarrow{finger\ intercept\ wood}) &= 0.4735
\end{aligned}$$

For the case when *"tap"* is disambiguated to its 'intercept' meaning, we do not yield the expected cosine correlations. For instance, *"police ..."* is not semantically closer to *"police intercept ..."* than to *"police knock ..."*, as one would expect. This might be since policemen knock many objects, such as tables and doors, and also since *tap* is too strongly associated with its knocking meaning than with its intercepting meaning.

$$\begin{aligned}
\cos(\overrightarrow{police\cdots}, \overrightarrow{police\ knock\cdots}) &= 0.0740 \\
\cos(\overrightarrow{police\cdots}, \overrightarrow{police\ intercept\cdots}) &= 0.0599 \\
\cos(\overrightarrow{police\ tap\cdots}, \overrightarrow{police\ knock\cdots}) &= 0.6630 \\
\cos(\overrightarrow{police\ tap\cdots}, \overrightarrow{police\ intercept\cdots}) &= 0.4597 \\
\cos(\overrightarrow{police\ tap\ phone}, \overrightarrow{police\ knock\ phone}) &= 0.6662 \\
\cos(\overrightarrow{police\ tap\ phone}, \overrightarrow{police\ intercept\ phone}) &= 0.4954
\end{aligned}$$

Because of these individual mismatches, we require a larger scale evaluation to get a more general picture, which we perform in the following section.

5.1 Larger-Scale Evaluation

We apply this method for incremental disambiguation in the full versions of the above mentioned datasets to see how well it scales up. Previous work on compositional distributional semantics provides three preliminary datasets suitable for this task: in each, sets of transitive S-V-O sentences in which the verb V is ambiguous are paired with human judgements of similarity between each given sentence and two possible paraphrases (e.g. for the sentence *"footballer dribbles ball"*, the possible paraphrases 'footballer carries ball' and 'footballer drips ball'). Grefenstette and Sadrzadeh (2011) provide a dataset with 32 paraphrase examples (hereafter GS2011); Grefenstette and Sadrzadeh (2015) a modification and extension of this to 97 paraphrase examples (GS2012); and Kartsaklis et al. (2013a) a further 97 examples on a different verb set (KSP2013).[3]

The GS2011 dataset is small, and contains judgements from only 12 annotators per example; the authors found it not to show significant differences between additive baselines and more complex compositional methods. The extended GS2012 version provides a larger set of 97 examples, each with 50 annotators' judgements; we expect it to provide a more reliable test. KSP2013 is then the same size, but selects the verbs using a different method. While Grefenstette and Sadrzadeh (2015) chose verbs which spanned multiple senses in WordNet (Fellbaum, 1998), taking the paraphrases as two of their most distant senses, Kartsaklis et al. (2013a) chose verbs specifically for their ambiguity, based on psycholinguistic evidence collected by eye tracking and human evaluation by Pickering and Frisson (2001). We therefore expect the KSP2013 dataset to provide an evaluation which is not only robust but a more direct test of the task of disambiguation in natural dialogue.

Again, we use the same 300-dimensional `word2vec` vectors and 300x300-dimensional verb tensors derived from them. For sentence composition, we

[3]Note that despite the date of the associated publication (Grefenstette and Sadrzadeh, 2015), the GS2012 dataset was created in 2012 and came second in the series. All datasets are publicly available; we provide information on how to download them, together with the software used here for our experiments, for replication purposes at `https://osf.io/hby4e/`.

now compare the method used in the previous section, from (Grefenstette and Sadrzadeh, 2011), which we term "G&S" below; with alternatives proposed by Kartsaklis et al. (2013b) termed "copy-subj" and "copy-obj". Here, $\odot$ denotes pointwise multiplication and $\otimes$ the Kronecker product as before, and $\times$ denotes matrix multiplication:

$$\begin{aligned}
\text{G\&S} : \overrightarrow{A} &= \overrightarrow{V} \odot (\overrightarrow{S} \otimes \overrightarrow{O}) \\
\text{copy-subj} : \overrightarrow{A} &= \overrightarrow{S} \odot (\overrightarrow{V} \times \overrightarrow{O}) \\
\text{copy-obj} : \overrightarrow{A} &= \overrightarrow{O} \odot (\overrightarrow{V}^T \times \overrightarrow{S})
\end{aligned}$$

The latter alternatives have been shown to perform better in some compositional tasks (see e.g. Kartsaklis et al., 2013b; Milajevs et al., 2014). We also compare the use of the identity I and sum T^+ to represent nodes with unsatisfied requirements; given our disambiguation task setting here, the natural way to use the direct sum $T^\oplus$ is to average the resulting distances over its output tuples, thus making it effectively equivalent to using the sum in this case. We compare these options to a simple, but often surprisingly effective, additive baseline (Mitchell and Lapata, 2008): summing the vectors for the words in the sentence. In this case, verbs are represented by their `word2vec` vectors, just as nouns (or any other words) are, viz. without taking their grammatical role into account; and incremental results are simply the sum of the words seen so far.

We evaluate the accuracy of these approaches by comparing to the human judgements in terms of the direction of preference indicated for the two possible paraphrases.[4] As several human judges were used for each sentence, we compare to the mean judgement for each sentence-paraphrase pair. Accuracy can therefore be calculated directly in terms of the percentage of sentences for which the most similar paraphrase is correctly identified. Given our incremental setting, we can make this comparison at three points in each S-V-O sentence (after parsing the subject S only; after parsing S and V; and after parsing the full S-V-O), at each point comparing the similarity between the (partial) sentence and each of the (partial) paraphrase sentences. Note though that after parsing S only, all methods are equivalent: the only information available is the vector representing the subject noun, the ambiguous verb has not even been observed, and disambiguation is therefore a random choice with 50% accuracy; the performance then diverges at S-V and S-V-O points.

Results Results for the small Grefenstette and Sadrzadeh (2011) dataset are shown in Figure 5; while none of our compositional approaches beat the additive baseline, it appears that the incremental performance after S-V may be reasonable compared to the full-sentence performance S-V-O. However, none of the

[4] We do not attempt to evaluate whether the *magnitude* of the preference matches the magnitude of human preferences, but only whether the direction is correct: in other words, we treat this as a classification rather than a regression task.

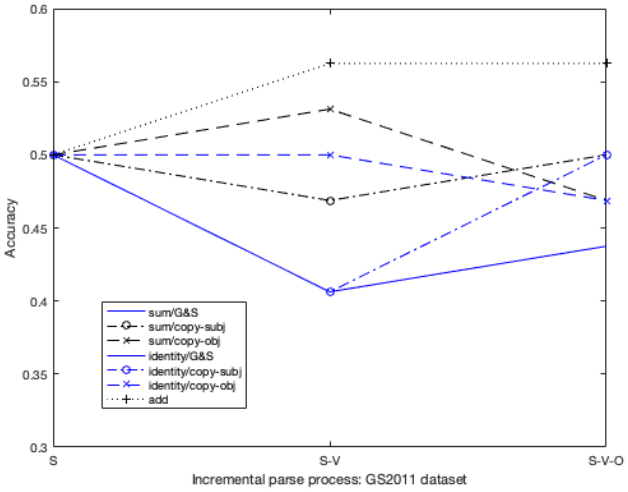


Figure 5: Mean disambiguation accuracy over the GS2011 dataset (Grefenstette and Sadrzadeh, 2011), as incremental parsing proceeds left-to-right through "S V O" sentences. Note that the *sum/G&S* and *identity/G&S* methods give identical average accuracy on this dataset, and thus share a line on the graph.

differences are statistically significant (a χ^2 test shows $\chi^2_{(1)} = 1.56, p = 0.21$ for the largest difference, *identity/G&S* vs. *add* at the S-V point), given the small size of the dataset, and conclusions are therefore hard to draw. One thing that, however, stands out, is that disambiguation accuracy increases from S-V to S-V-O for the relational G&S model and the copy-subject model. The additive model stays almost the same after adding the verb and after adding the object, while the copy-object method gets worse; these may be undesirable properties in terms of providing a good model of incrementality.

For the larger datasets, results are shown in Tables 1, 2 and depicted in Figures 6, 7. For GS2012, all methods do significantly better than chance (taking $p < 0.05$ for significance, $\chi^2_{(1)} = 5.08, p = 0.024$ for the worst method, *add*); the compositional methods outperform the additive baseline, and although the improvement is not statistically significant at the $p < 0.05$ level it suggests an effect ($p < 0.15$, with $\chi^2_{(1)} = 2.51, p = 0.11$ for the best method, *identity/copy-obj* at the V-O point). The copy-object method seems to do best, outperforming copy-subject and the G&S method, and particularly to perform well incrementally at the mid-sentence S-V point (76% accuracy, with 72% after S-V-O). Again, similar to GS2011, and despite the fact that copy-object does best on the overall accuracy, the *identity/G&S* and *identity/copy-subj* models seem to do best in terms of incremental accuracy development; their accuracies increase more when going from S-V to S-V-O, and seem to increase more smoothly through

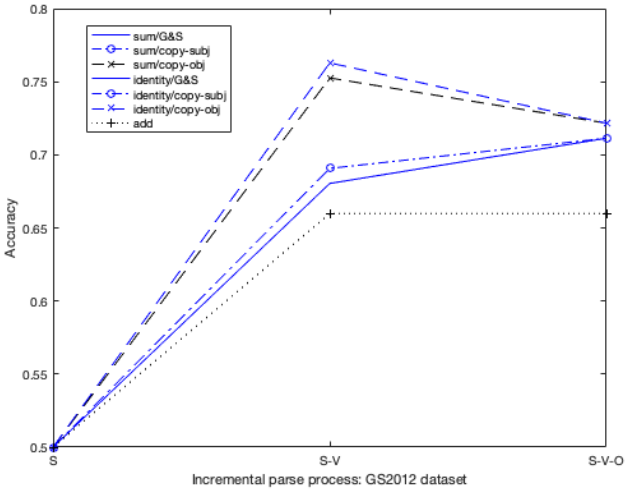


Figure 6: Mean disambiguation accuracy over the GS2012 dataset (Grefenstette and Sadrzadeh, 2015), as incremental parsing proceeds left-to-right through "S V O" sentences. Note that the *sum/G&S* and *identity/G&S* methods give identical average accuracy on this dataset, as do the *sum/copy-subj* and *identity/copy-subj* methods, and thus those pairs share lines on the graph.

the sentence, whereas the copy-obj models increase to S-V and then decrease.

For KSP2013, the task seems harder: here, the additive baseline performs almost at chance level with about 52%, but all the tensor-based compositional methods do better; the best improvement being significant at $p < 0.1$, although not at $p < 0.05$ ($\chi^2_{(1)} = 2.77, p = 0.096$ for *identity/copy-obj* at S-V-O). Again, the copy-object composition method seems to perform best, giving good accuracy at S-V and S-V-O points (62% accuracy); the G&S method does better this time, particularly at the mid-sentence point; but copy-subject does well for the full sentence but not incrementally. Copy-object with identity, the model that provides the best accuracy, also shows a steady increase in accuracy through the sentence, although copy-subject with identity still shows the steepest increase from S-V to S-V-O. This latter method shows the steepest increase in all the datasets.

Accuracy comparisons between the identity and sum/direct sum methods show little difference. As we see in Tables 1 and 2, whenever there is a difference in results among the different requirement representations, the identity approach gives slightly higher accuracy. An explanation of this is that the identity is only used as a mechanism to be able to compute a sentence representation in a compositional way, but without contributing information by itself. On the contrary, the sum and direct sum methods introduce averages of vectors found

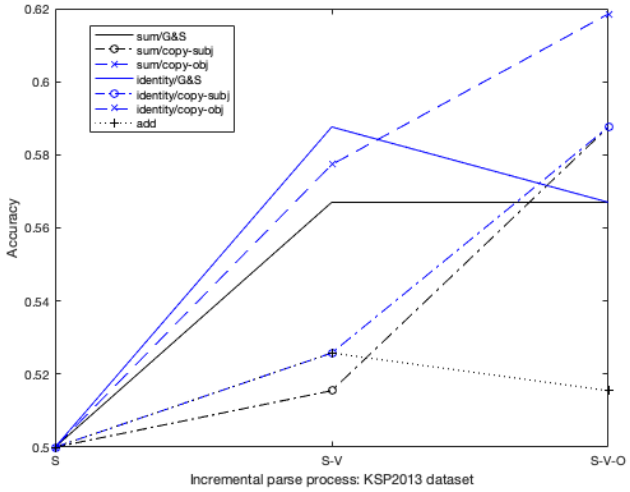


Figure 7: Mean disambiguation accuracy over the KSP2013 dataset (Kartsaklis et al., 2013a), as incremental parsing proceeds left-to-right through "S V O" sentences. Note that the *sum/copy-obj* and *identity/copy-obj* methods give identical average accuracy on this dataset, and thus share a line on the graph.

in the corpus, which is akin to adding noisy information to the sentence representation; remember that the datasets we use here (from which we must take our information about the possible continuations that we average over) are very small compared to the large corpora used to build standard word vectors. It is encouraging that all methods perform well; it may be that in larger datasets the sum methods will improve, given more information about the possible distributions over continuations, and in other tasks which depend on more than just average sentence distances, the sum and direct sum methods will diverge.

Discussion and comparison The main point of interest here, of course, is the intermediate point after processing S-V (but before seeing the object O): here the additive baseline does approximately as well as with full sentences, suggesting that most disambiguating information comes from the verb vector in these datasets. The compositional tensor-based methods on the other hand, particularly copy-object, seem able to use information from the combination of S-V to improve on that, and then to incorporate further information from O to improve again (at least with KSP2013). Composition therefore allows useful information from all arguments to be included; and it seems that our method allows that to be captured incrementally as the sentence proceeds.

An error analysis showed that in the majority of cases our overall best performing models (*sum/copy-obj, identity/copy-obj*) either correctly disam-

Composition Method	Representation of Requirements	Accuracy S	S+V	S+V+O
Addition	(N/A)	0.500	0.660	0.660
G&S	Identity	0.500	0.680	0.711
	Sum / Direct Sum	0.500	0.680	0.711
Copy-Sbj	Identity	0.500	0.691	0.711
	Sum / Direct Sum	0.500	0.691	0.711
Copy-Obj	Identity	0.500	0.763	0.722
	Sum / Direct Sum	0.500	0.753	0.722

Table 1: Mean disambiguation accuracy over the GS2011 dataset (Grefenstette and Sadrzadeh, 2011), as incremental parsing proceeds left-to-right through "S V O" sentences

Composition Method	Representation of Requirements	Accuracy S	S+V	S+V+O
Addition	(N/A)	0.500	0.526	0.515
G&S	Identity	0.500	0.588	0.567
	Sum / Direct Sum	0.500	0.567	0.567
Copy-Sbj	Identity	0.500	0.526	0.588
	Sum / Direct Sum	0.500	0.515	0.588
Copy-Obj	Identity	0.500	0.577	0.619
	Sum / Direct Sum	0.500	0.577	0.619

Table 2: Mean disambiguation accuracy over the KSP2013 dataset (Kartsaklis et al., 2013a), as incremental parsing proceeds left-to-right through "S V O" sentences

biguated both the S-V and the S-V-O pairs, or got it wrong in both cases; in other words, the incremental accuracy was as good (or bad) as that for complete sentences. In a minority of cases, though, the incremental behaviour diverged (either S-V was disambiguated correctly, while S-V-O was not, or vice versa). These are the cases of interest here (for discussion of the behaviour of different compositional models for full sentences, see Kartsaklis et al., 2013b).

Interestingly, a prominent erratic ambiguous verb was *to file*, where in some cases, the *smooth* meaning was expected but the model wrongly computed it to be the *register* meaning, and in the other cases, the *register* meaning was expected whereas the model wrongly computed it to be the *smooth* meaning. Examples of the data set entries were (all words in stem form):

(1) *woman file nail*
englishman file steel

(2) *state file declaration*
union file lawsuit

where the *smooth* meaning is expected in (1) and *register* in (2). For (1) examples, the copy-obj models predicted correctly at the S-V point, and then incorrectly at the S-V-O point. These seem to be examples where most disambiguating information intuitively comes in the object. We therefore suspect that although the S-V subject and verb tensor combination itself contains sufficient information about the kind of object in these cases (see Kartsaklis et al. 2013b for discussion of how the copy-obj method encodes more object information), these particular objects did not occur frequently enough in the corpus with this verb meaning, but had more occurrences in the context of other verbs. In the case of *file nail*, for instance, the noun *nail* may occur more with verbs such as *to hammer* or *to sell*, or *to cut*, rather than the verb *to file*. The copy-subj models performed the opposite way, predicting incorrectly at S-V and then correctly with the full S-V-O sentence: here, the S-V composition themselves encode less information about the disambiguating object (hence incorrectness at S-V), and this can be supplied later on S-V-O composition, while giving the object less weight than with the copy-obj method.

We observed the same pattern for our most smoothly incremental models: copy-subject with sum and identity. In the majority of cases, these models either got the meaning of the verb correctly for both S-V and S-V-O, or got it wrong, again for both S-V and S-V-O. Their mistakes, i.e. cases where S-V was correctly disambiguated, but S-V-O was not, were more varied, apart from the verb *to file*, they also had instances of *to cast*, *to tap* and *to lace*, in the following contexts:

(3) *company file account*
boat cast net
palace cast net
monitor tap conversation
child lace shoe

In all of these cases, the object provided in the data set has occurred more frequently with contexts of other verbs, e.g. *account* in the first sentence above has occurred more in the context of verbs such as *funded* or *issued*; *net* in the second and third examples is itself ambiguous and occurred much more frequently in its financial sense (where it contrasts with *gross*) in the very large naturally occurring dataset taken as base. Similarly for *conversation* and *shoe*, which occurred more with *had* and *wore* respectively, than *tapped* and *laced*.

Differences between the sum and identity methods are smaller and thus harder to investigate in a conclusive manner. Some verbs, such as *dribble*, show interesting differences: for *woman dribble wine*, identity seems to give better accuracy at the S-V stage than at S-V-O; for *player dribble ball* it is the opposite.

Overall, following the Kartsaklis et al. (2013b) demonstration that copy-obj outperforms others for full-sentence disambiguation in virtue of encoding more information about the object, the results here, which incorporate in addition an incrementality factor, also indicate that copy-obj does better overall, and for similar reasons, though here based on probability rather than encoding.

However, with some verbs getting disambiguated with their objects better than with their subject and some verbs the other way round, it is hard to evaluate which model's performance is really most desirable. In future work we would hope to investigate comparisons with human ratings of disambiguation at the S-V stage, but this raises complex questions about datasets and about bias in the vector/tensor corpora which are beyond the scope of this paper.

6 Discussion

Although the theoretical predictions of the model have only been verified on S-V-O triples, they are immediately applicable to sentences of greater complexity. Of importance here, however, are utterances arising within natural dialogue, and of those, particularly unfinished and interrupted instances. These kinds of utterance have not been dealt with in the commonly used type-logical vector space approaches so far, as those rely on a sentential level of grammaticality. As our simple experiment shows, our setting does not rely on sentential grammaticality: we have theoretically prescribed how to build vector representations for any DS tree; on the practical side, we have applied these prescriptions to subject-only, subject-verb, and subject-verb-object strings. This is the first time it has been shown that disambiguation of unfinished utterances can be computed incrementally in vector space semantics, not only opening the practical possibilities of real-time distributional semantic processing for spoken Natural Language Understanding tasks, but also allowing for a more realistic simulation of human processing than previously possible. The match that our setting provides for human disambiguation judgements is being derived solely on the basis of observed co-occurrences between words and syntactic roles in a corpus, without any specification of content intrinsic to the word itself. Further experiments will be needed to extend this approach to larger datasets and to dialogue data and examine its effectiveness, perhaps using the work extending DS grammars to dialogue (Eshghi et al., 2017), and possibly evaluating on the similarity dataset of Wijnholds and Sadrzadeh (2019) that extends the transitive sentence datasets used in this paper to a verb phrase elliptical setting.

Our assumption from the outset of this work was that distributions across a sufficiently large corpus can be taken to provide an analogue and basis for formal modelling of the observation that interpretation of words depends on contingent, contextual and encyclopaedic facts associated with objects. To place these results and the adopted methodology in a psychological perspective, the way in which these statistical methods show that discrete facets of meaning of an individual word are progressively distinguishable in an incremental way provides at least partial confirmation that the meaning that words have is recoverable from *affordances* made available in the contingent contexts in which they occur, these being anticipations routinely associated with the word in question over many uses that they come to constitute, including the actions triggered by the word.[5] Moreover, the underlying concept of a context of affordances has the

[5]The original Gibsonian concept of affordance, 'perceivable relations between an organism's

cross-temporal, cross-spatial attributes shared by "big-data" corpora.

We thus take the results as provisionally confirming a thin concept of meaning, not associated with some intrinsically fixed encoded content, but merely a non-deterministic set of associations which the word triggers for the individual agent(s). We also expect to be able to deal with cases when an interpretation shifts during the incremental process (say, when uttering "The footballer dribbled beer down his chin"), when the incoming input acts as a filter over-riding an otherwise accumulating default. This is exactly what one would expect of an account with a basis in non-deterministic meanings, the underpinnings allowing variability as the interpretation gradually consolidates, directly in line with a range of Radical Embodied Cognition perspectives (Clark, 2016; Bruineberg and Rietveld, 2014; Kempson and Gregoromichelaki, 2019). It also gives us hope that such an approach (although we currently have no direct model of this) should extend to modelling the more general shifts in understanding that occur within the ubiquitous coordinating to-and-fro between interlocutors in dialogue (Healey et al., 2018). In the mean time, we hope these provisional results make a contribution towards grounding the claim that languages are defined as procedures for inducing growth of specifications of content in real time, with plasticity of such constituent parts playing an irreducible role.

Acknowledgements

The contributions of this paper reflect joint work with a number of people over a considerable period. In particular we thank Arash Eshghi for extensive illuminating discussions during the writing of this paper and of other related work in this line of research, and Eleni Gregoromichelaki for ongoing insights into the relevance of the issues raised for the larger cognitive perspective. Special thanks go to the anonymous reviewers of this paper, and the editors, for helpful comments which led to substantial improvements. Purver is partially supported by the EPSRC under grant EP/S033564/1, and by the European Union's Horizon 2020 program under grant agreements 769661 (SAAM, Supporting Active Ageing through Multimodal coaching) and 825153 (EMBEDDIA, Cross-Lingual Embeddings for Less- Represented Languages in European News Media). The results of this publication reflect only the authors' views and the Commission is not responsible for any use that may be made of the information it contains.

This is a post-peer-review, pre-copyedit version of an article published in the Journal of Logic, Language and Information. The final authenticated version will be available online at: `http://dx.doi.org/TBD`.

abilities and the properties of the environment' (Anderson, 2014), was restricted to that of affordances for motor activity made available by the environment to the individual in question, but following Bruineberg and Rietveld *inter alia* we take affordances to be all types of possibility relevant to an agent for action within the environment provided (Clark, 2016; Bruineberg and Rietveld, 2014; Rietveld et al., 2018), including words and the grammar.

References

Anderson, M. L. (2014). *After Phrenology: Neural Reuse and the Interactive Brain*. MIT Press, Cambridge, MA.

Baroni, M., Bernardi, R., and Zamparelli, R. (2014a). Frege in space: A program of compositional distributional semantics. *Linguistic Issues in Language Technology*, 9.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The WaCky Wide Web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Baroni, M., Dinu, G., and Kruszewski, G. (2014b). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 238–247, Baltimore, Maryland. Association for Computational Linguistics.

Bruineberg, J. and Rietveld, E. (2014). Self-organization, free energy minimization, and optimal grip on a field of affordances. *Directions in Human Neuroscience*, 8(599).

Burnard, L. (2000). *Reference Guide for the British National Corpus (World Edition)*. Oxford University Computing Services, Oxford, UK.

Cann, R., Kempson, R., and Marten, L. (2005). *The Dynamics of Language: An Introduction. Syntax and Semantics. Volume 35*. Academic Press, San Diego, CA.

Cann, R., Kempson, R., and Purver, M. (2007). Context and well-formedness: The dynamics of ellipsis. *Research on Language and Computation*, 5(3):333–358.

Carston, R. (2019). Ad-hoc concepts, polysemy and the lexicon. In *Relevance, Pragmatics and Interpretation*, pages 150–162. Cambridge University Press, Cambridge, UK.

Clark, A. (2016). *Surfing Uncertainty: Prediction, Action and the Embodied Mind*. Oxford University Press, Oxford, UK.

Clark, H. H. (1996). *Using Language*. Cambridge University Press, Cambridge, UK.

Clark, S. (2013). Vector space models of lexical meaning. In Heunen, C., Sadrzadeh, M., and Grefenstette, E., editors, *Quantum Physics and Linguistics: A Compositional, Diagrammatic Discourse*, pages 359–377. Oxford University Press, Oxford, UK, 1st edition.

Clark, S., Coecke, B., and Sadrzadeh, M. (2013). The Frobenius anatomy of relative pronouns. In *13th Meeting on Mathematics of Language (MoL)*, pages 41–51, Stroudsburg, PA. Association for Computational Linguistics.

Coecke, B. (2019). The mathematics of text structure. *Computing Research Repository (CoRR)*, abs/1904.03478.

Coecke, B., Grefenstette, E., and Sadrzadeh, M. (2013). Lambek vs Lambek: Functorial vector space semantics and string diagrams for Lambek calculus. *Annals of Pure and Applied Logic*, 164(11):1079–1100.

Coecke, B., Sadrzadeh, M., and Clark, S. (2010). Mathematical foundations for a compositional distributional model of meaning. *Linguistic Analysis*, 36:345–384.

Cooper, R. (2005). Records and record types in semantic theory. *Journal of Logic and Computation*, 15(2):99–112.

Curran, J. (2004). *From Distributional to Semantic Similarity*. PhD thesis, School of Informatics, University of Edinburgh.

Eshghi, A., Howes, C., Gregoromichelaki, E., Hough, J., and Purver, M. (2015). Feedback in conversation as incremental semantic update. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 261–271, London, UK. Association for Computational Linguistics.

Eshghi, A., Shalyminov, I., and Lemon, O. (2017). Bootstrapping incremental dialogue systems from minimal data: The generalisation power of dialogue grammars. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2220–2230, Copenhagen, Denmark. Association for Computational Linguistics.

Fellbaum, C., editor (1998). *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Firth, J. (1957). A synopsis of linguistic theory 1930–1955. In *Studies in Linguistic Analysis*. Blackwell, Oxford, UK.

Fretheim, T. (2019). The polysemy of a Norwegian modal adverb. In *Relevance, Pragmatics and Interpretation*, pages 163–173. Cambridge University Press, Cambridge, UK.

Grefenstette, E. and Sadrzadeh, M. (2011). Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1394–1404, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Grefenstette, E. and Sadrzadeh, M. (2015). Concrete models and empirical evaluations for the categorical compositional distributional model of meaning. *Computational Linguistics*, 41(1):71–118.

Healey, P. G., Mills, G. J., Eshghi, A., and Howes, C. (2018). Running repairs: Coordinating meaning in dialogue. *Topics in Cognitive Science*, 10(2):367–388.

Hough, J. and Purver, M. (2012). Processing self-repairs in an incremental type-theoretic dialogue system. In *Proceedings of the 16th SemDial Workshop on the Semantics and Pragmatics of Dialogue (SeineDial)*, pages 136–144, Paris, France.

Hough, J. and Purver, M. (2017). Probabilistic record type lattices for incremental reference processing. In Chatzikyriakidis, S. and Luo, Z., editors, *Modern Perspectives in Type-Theoretical Semantics*, pages 189–222. Springer International Publishing, Basel, Switzerland.

Howes, C., Purver, M., Healey, P. G. T., Mills, G. J., and Gregoromichelaki, E. (2011). On incrementality in dialogue: Evidence from compound contributions. *Dialogue and Discourse*, 2(1):279–311.

Hupkes, D., Bouwmeester, S., and Fernández, R. (2018). Analysing the potential of seq-to-seq models for incremental interpretation in task-oriented dialogue. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 165–174, Brussels, Belgium. Association for Computational Linguistics.

Kartsaklis, D. (2015). *Compositional Distributional Semantics with Compact Closed Categories and Frobenius Algebras*. PhD thesis, Department of Computer Science, University of Oxford.

Kartsaklis, D. and Sadrzadeh, M. (2013). Prior disambiguation of word tensors for constructing sentence vectors. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 1590–1601.

Kartsaklis, D., Sadrzadeh, M., and Pulman, S. (2013a). Separating disambiguation from composition in distributional semantics. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning (CoNLL)*, pages 114–123, Sofia, Bulgaria.

Kartsaklis, D., Sadrzadeh, M., Pulman, S., and Coecke, B. (2013b). Reasoning about meaning in natural language with compact closed categories and Frobenius algebras. In *Logic and Algebraic Structures in Quantum Computing and Information*. Cambridge University Press, Cambridge, UK.

Kempson, R., Cann, R., Gregoromichelaki, E., and Chatzikyriakidis, S. (2016). Language as mechanisms for interaction. *Theoretical linguistics*, 42(3-4):203–276.

Kempson, R., Cann, R., Gregoromichelaki, E., and Purver, M. (2015). Ellipsis. In *Handbook of Contemporary Semantic Theory*, pages 156–194. Blackwell, Oxford, UK, 2nd edition.

Kempson, R. and Gregoromichelaki, E. (2019). Procedural syntax. In *Relevance, Pragmatics and Interpretation*, pages 187–202. Cambridge University Press, Cambridge, UK.

Kempson, R., Gregoromichelaki, E., and Howes, C., editors (2011). *The Dynamics of Lexical Interfaces*. CSLI, Chicago, IL.

Kempson, R., Meyer-Viol, W., and Gabbay, D. (2001). *Dynamic Syntax: The Flow of Language Understanding*. Blackwell, Oxford, UK.

Krishnamurthy, J. and Mitchell, T. M. (2013). Vector space semantic parsing: A framework for compositional vector space models. In *Proceedings of the ACL Workshop on Continuous VSMs and their Compositionality*.

Lambek, J. (1958). The mathematics of sentence structure. *American Mathematics Monthly*, 65:154–170.

Lambek, J. (1997). Type grammars revisited. In *Proceedings of the 2nd International Conference on Logical Aspects of Computational Linguistics (LACL)*. Springer.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 17th International Conference on Computational Linguistics (COLING)*, pages 768–774. Association for Computational Linguistics.

Maillard, J., Clark, S., and Grefenstette, E. (2014). A type-driven tensor-based semantics for CCG. In *Proceedings of the Type Theory and Natural Language Semantics Workshop, EACL 2014*.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 746–751.

Milajevs, D., Kartsaklis, D., Sadrzadeh, M., and Purver, M. (2014). Evaluating neural word representations in tensor-based compositional settings. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 708–719, Doha, Qatar. Association for Computational Linguistics.

Mitchell, J. and Lapata, M. (2008). Vector-based models of semantic composition. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 236–244.

Mitchell, J. and Lapata, M. (2010). Composition in distributional models of semantics. *Cognitive Science*, 34:1388–1439.

Moortgat, M. and Wijnholds, G. (2017). Lexical and derivational meaning in vector-based models of relativisation. In *Proceedings of the 21st Amsterdam Colloquium*.

Muskens, R. and Sadrzadeh, M. (2016). Context update for lambdas and vectors. In *LNCS Proceedings of the 9th International Conference on Logical Aspects of Computational Linguistics*, Nancy. Springer.

Partee, B. (2018). Changing notions in the history of linguistic competence. In *The Science of Meaning: Essays on the Metatheory of Natural Language Semantics*, pages 172–186. Oxford University Press, Oxford, UK.

Pickering, M. and Frisson, S. (2001). Processing ambiguous verbs: Evidence from eye movements. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 27:556–573.

Purver, M., Eshghi, A., and Hough, J. (2011). Incremental semantic construction in a dialogue system. In *Proceedings of the 9th International Conference on Computational Semantics*, IWCS '11, pages 365–369, Stroudsburg, PA, USA. Association for Computational Linguistics.

Purver, M., Gregoromichelaki, E., Meyer-Viol, W., and Cann, R. (2010). Splitting the 'I's and crossing the 'You's: Context, speech acts and grammar. In *Proceedings of the 14th SemDial Workshop on the Semantics and Pragmatics of Dialogue*, pages 43–50.

Purver, M., Hough, J., and Gregoromichelaki, E. (2014). Dialogue and compound contributions. In Stent, A. and Bangalore, S., editors, *Natural Language Generation in Interactive Systems*, pages 63–92. Cambridge University Press, Cambridge, UK.

Recanati, F. (2017). Contextualism and polysemy. *Dialectica*, 71(3):379–397.

Rietveld, E., Denys, D., and van Westen, M. (2018). Ecological-enactive cognition as engaging with a field of relevant affordances: the skilled intentionality framework (SIF). In Newen, A., de Bruin, L., and Gallagher, S., editors, *The Oxford Handbook of 4E Cognition*, pages 156–194. Oxford University Press, Oxford, UK.

Rubenstein, H. and Goodenough, J. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Sadrzadeh, M., Clark, S., and Coecke, B. (2013). Frobenius anatomy of word meanings I: subject and object relative pronouns. *Journal of Logic and Computation*, 23:1293–1317.

Sadrzadeh, M., Clark, S., and Coecke, B. (2014). Frobenius anatomy of word meanings II: possessive relative pronouns. *Journal of Logic and Computation*, 26:785–815.

Sadrzadeh, M., Purver, M., Hough, J., and Kempson, R. (2018a). Exploring semantic incrementality with Dynamic Syntax and vector space semantics. In *Proceedings of the 22nd SemDial Workshop on the Semantics and Pragmatics of Dialogue (AixDial)*, pages 122–131, Aix-en-Provence.

Sadrzadeh, M., Purver, M., and Kempson, R. (2017). Incremental distributional semantics for Dynamic Syntax. In *Proceedings of the 1st Dynamic Syntax Conference*, London, UK.

Sadrzadeh, M., Purver, M., and Kempson, R. (2018b). A tensor-based vector space semantics for Dynamic Syntax. In *Proceedings of the 2nd Dynamic Syntax Conference*, Edinburgh, UK.

Schegloff, E. (1984). On some questions and ambiguities in conversation. In *Structures of Social Action: Studies in Conversation Analysis*, pages 28–52. Cambridge University Press, Cambridge, UK.

Schütze, H. (1998). Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.

Vinyals, O. and Le, Q. (2015). A neural conversational model. In *Proceedings of the 31st International Conference on Machine Learning (ICML) Deep Learning Workshop*.

Wijnholds, G. and Sadrzadeh, M. (2019). Evaluating composition models for verb phrase elliptical sentence embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics.

Wijnholds, G. J. (2017). Coherent diagrammatic reasoning in compositional distributional semantics. In *Proceedings of the 24th Workshop on Logic, Language, Information and Computation (WoLLIC)*, pages 371–386.