Graded Similarity in Context

Tim Lawson

December 14, 2023

1 Introduction

In his *Foundations of Arithmetic*, Frege promises "never to ask for the meaning of a word in isolation, but only in the context of a proposition" (1980, p. xvii). This 'context principle' is intuitive: words are frequently polysemous, or assume different connotations and emphasis within different expressions. Historically, however, contextuality has been a problem for distributional meaning representations. Founded on the distributional hypothesis (Harris 1954; Firth 1957), both count-based and predictive models of word meaning[1] originally produced a single representation for each word in the model's vocabulary. One of these *static* representations must, therefore, encode all of a word's senses and connotations, which is an obstacle to its use in modelling context-dependent phenomena.

Prior to the widespread availability of pre-trained word embeddings (e.g., Mikolov, Chen, et al. 2013; Pennington et al. 2014) and their successors, this problem was generally addressed by one of two approaches: firstly, by producing a representation for each sense of a target word and disambiguating between them in the given context (*word-sense disambiguation*); or secondly, by composing the representation of the target word with the representations of the words in its context (*contextualisation*). These approaches have been largely overshadowed by the advent of model architectures that take sequences as inputs and naturally produce *contextual* representations of the items in the sequence, such as Transformers (Vaswani et al. 2017).

To my knowledge, however, there has been scant direct comparison of the performance of these contextual representations with the application of prior methods of contextualisation to static representations. SemEval-2020 Task 3, "Graded Word Similarity in Context" (Armendariz, Purver, Pollak, et al. 2020), presents an opportunity to make such a comparison. Briefly, the task is to predict the human judgment of similarity of the same pair of words in two different contexts (Section 2). I elected to focus on the first subtask, which is to predict the *change* in similarity, rather than the absolute similarity in each context. Specifically, I evaluated the results obtained by computing the cosine similarity between static and contextual embeddings and the composition of these embeddings within a fixed-size context window.

2 Task definition

The first subtask of SemEval-2020 Task 3 is to predict the direction and magnitude of the change in the human judgment of similarity of the same pair of target words in two different contexts. The task is unsupervised: the CoSimLex dataset was used to evaluate the task submissions but only a minimal 'practice kit' of fewer than ten instances was provided in advance. CoSimLex extends the well-known SimLex-999 dataset (Hill et al. 2015) to include contexts for each pair in English (n = 340), Finnish (n = 24), Croatian (n = 112), and Slovene (n = 111) (Armendariz, Purver, Ulčar, et al. 2020, pp. 39–42).

The score for the first subtask was computed by the uncentered (zero-mean) Pearson correlation coefficient between the predicted changes in similarity and the human judgments represented in the

[1] This terminological distinction is due to Baroni et al. (2014).

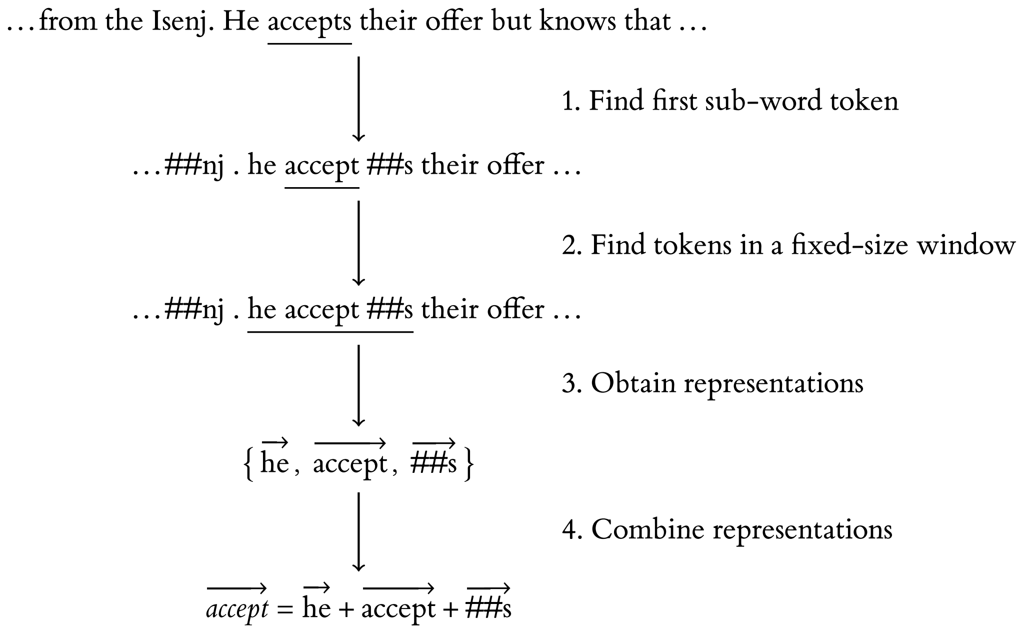

Figure 1: A schematic of the procedure used to obtain a contextualised representation of a target word from pre-trained static or contextual embeddings. In this example, the target word is "accept", the window size is three, and the composition operation is addition.

CoSimLex dataset (Armendariz, Purver, Ulčar, et al. 2020, p. 42). This metric is equivalent to the cosine similarity between the two vectors of results:

$$\text{score}(\vec{\hat{y}}, \vec{y}) = \frac{\sum_{i=1}^{n} \hat{y}_i y_i}{\left(\sum_{i=1}^{n} \hat{y}_i^2\right)\left(\sum_{i=1}^{n} y_i^2\right)} = \frac{\vec{\hat{y}} \cdot \vec{y}}{\|\vec{\hat{y}}\| \|\vec{y}\|} \tag{1}$$

A consequence of this choice of evaluation metric is that composing representations by addition and their arithmetic mean produce the same results; hence, I elected only to evaluate the simpler operatoin of addition among the two.

3 Related work

Additive composition (e.g. Kintsch 2001; Mitchell and Lapata 2008; Mikolov, Sutskever, et al. 2013). Analysis of the effect of window size on static embeddings. Analysis of contextual embeddings.

Contextual-embedding models have found broad success on benchmarks that involve language understanding. A survey of contextual embeddings is given by Liu et al. (2020). However, as Arora et al. (2020) point out, these models require significantly greater computational resources relative to static-embedding models. THe authors demonstrated that static embeddings 'perform surprisingly well' on named-entity recognition and sentiment analysis tasks when there is adequate training data and it is linguistically simple. Furthermore, Gupta et al. (2019) and Bommasani et al. (2020) have demonstrated that static-embeddings can be obtained from contextual-embedding models that outperform prior static models while retaining their computational advantages.

4 Methodology

Originally, it would not have been possible to optimise a parameterised model for the task except by reference to a separate dataset; therefore, I chose to focus on the application of pre-trained static and contextual embeddings. The basic procedure of the methods that I evaluated is as follows. For each

Model name	English	Finnish	Croatian	Slovene
`EMBEDDIA/crosloengual-bert`	✓	✓	✓	✓
`bert-base-cased`	✓			
`bert-base-multilingual-cased`	✓	✓	✓	✓
`bert-base-multilingual-uncased`	✓	✓	✓	✓
`bert-base-uncased`	✓			
`bert-large-cased`	✓			
`bert-large-cased-whole-word-masking`	✓			
`bert-large-uncased`	✓			
`bert-large-uncased-whole-word-masking`	✓			
`classla-bcms-bertic`			✓	
`TurkuNLP/bert-base-finnish-cased-v1`		✓		
`TurkuNLP/bert-base-finnish-uncased-v1`		✓		
`TurkuNLP/bert-large-finnish-cased-v1`		✓		

Figure 2: The pre-trained models available via the HuggingFace *Transformers* library (Wolf et al. 2020) that I chose to evaluate for each language.

pair of target words and each of the two contexts in which they appear, I obtained a contextualised representation of a target word by: finding the index of the target word's first sub-word token within the tokens of the target word's context; finding the tokens within a fixed-size window around the target word's first token; obtaining the embeddings of the tokens in the window; and combining the the embeddings to produce a single representation of the target word.

In all cases, the tokenization was performed by and the embeddings were obtained from pre-trained models available via the HuggingFace *Transformers* library (Wolf et al. 2020). The models that I evaluated for each language are given in Table ??. For the static-embedding variants of the procedure, I used the models' input embeddings; for the contextual-embedding variants, I used the models' outputs. Several of the submissions to SemEval-2020 Task 3 used a combination of the weights of a Transformer model's hidden states (e.g., Gamallo 2020, p. 276; Pessutto et al. 2020, p. 3; Hettiarachchi and Ranasinghe 2021, p. 4); a thorough comparison of the performance of variants of this approach is beyond the scope of this paper. Notably, the use of a sub-word vocabulary by these models (e.g., Devlin et al. 2019, p. 4174) dictates that a target word may be represented by a different number of tokens in each context. As a result, the representations of a pair of target words may be different in each context, even if they are static and the window size is zero. This explains the non-zero scores obtained by models of this kind (Section 8.2), particularly for the Finnish language.

Inspired by Kintsch (2001) and Mitchell and Lapata (2008), I predominantly investigated the application of element-wise addition and multiplication as composition operations. However, preliminary experiments indicated that multiplication performed poorly across all languages, models, and window sizes; hence, it was discarded before the final evaluation. Additionally, I chose to evaluate the concatenation ('stacking') of embeddings. In the case that the number of embeddings was fewer than the context-window size, i.e., the target word was close to the beginning or the end of its context, I right-padded the concatenated embedding with zeros to obtain contextual embeddings of equal length.

5 Results

5.1 Cost-benefit analysis of contextual embeddings

5.2 Language-specificity of window-size effects

Generally, I found that the scores obtained by both static- and contextual-embedding models were maximised by a non-zero context-window size. Due to the computational expense of exhaustively searching the possible window sizes, I applied a heuristic to constrain the search space. A naïve estimation of the average number of words in each context, i.e., segmenting on whitespace, gave a result of between 40 and 60 for the different languages. Therefore, for the static-embedding models, I chose 50 as an upper bound on the window size. The motivation to choose a smaller maximum window size for contextual-embedding models was similarly economical (Section 5.1); however, as the window size approaches the length of the sequence, one would expect a combination of token representations to be superseded by the sequence-level representation of the model, e.g., the special `CLS` token of BERT variants (Devlin et al. 2019, p. 4174). These heuristics were largely vindicated by the results of the evaluation, which demonstrated that the scores decrease as the window size approaches the maximum.

The influence of the window size is intuitive in the case of static embeddings. Without a context window, the representations of a target word only differ between expressions if the word is represented by different sub-word tokens in the different expressions. A similar argument applies to contextual embeddings, in that a target word may be represented by multiple sub-word tokens. The window sizes that maximise the score for each language and model are given in Table **??**.

6 Discussion

Batchkarov et al. (2016) critically analyse word similarity as an evaluation methodology for distributional semantic models. In particular, the notion of 'similarity' manifested by these models encompasses a broad range of semantic relations (e.g., Padó and Lapata 2003, p. 2), with the consequence that performance on an intrinsic word-similarity task does not necessarily translate to extrinsic downstream tasks (Batchkarov et al. 2016, pp. 7–8). Moreover, inter-annotator agreement is generally poor for word-similarity tasks in comparison to more specific downstream tasks (ibid., pp. 8–9).

7 Conclusion

In this paper, I have presented the results of a hypothetical submission to SemEval-2020 Task 3, "Graded Word Similarity in Context". The purpose of this evaluation was to compare the performance of static and contextual embeddings and their composition within a fixed-size context window on the task of predicting the change in the human judgment of similarity of a pair of words in two different contexts. I found that contextual embeddings generally outperformed static embeddings, but at a significant computational cost, and that composition benefited both static and contextual embeddings of sub-word tokens, with highly language-specific dependence on the window size.

The results that I have given must be interpreted in context: the original submission authors did not have access to the evaluation dataset prior to submitting their results, only the 'practice kit' of very few instances, and were therefore unable to optimise a parameter such as the window size prior to submission. Additionally, the models that I used were not necessarily available to the authors. With these caveats, I achieved several notable results:

- The contextual embeddings of `EMBEDDIA/crosloengual-bert` with a window size of three would have placed second among the Croatian submissions (0.708).
- The contextual embeddings of `TurkuNLP/bert-base-finnish-uncased-v1` with a window size of zero would have placed fourth among the Finnish submissions (0.679).
- The *static* embeddings of `TurkuNLP/bert-base-finnish-uncased-v1` with a window size of zero outperform several of the Finnish submissions, including the baseline (0.564).

Given the significant expense of applying contextual-embedding models, these results highlight the importance of analysing the complexity of the task at hand and considering the possibility that a simpler model produces adequate results.

References

Armendariz, Carlos Santos, Matthew Purver, Senja Pollak, et al. (2020). "SemEval-2020 Task 3: Graded Word Similarity in Context". In: *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. Barcelona (online): International Committee for Computational Linguistics, pp. 36–49.

Armendariz, Carlos Santos, Matthew Purver, Matej Ulčar, et al. (2020). *CoSimLex: A Resource for Evaluating Graded Word Similarity in Context*.

Arora, Simran et al. (2020). "Contextual Embeddings: When Are They Worth It?" In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by Dan Jurafsky et al. Online: Association for Computational Linguistics, pp. 2650–2663.

Baroni, Marco, Georgiana Dinu, and Germán Kruszewski (2014). "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 (Volume 1: Long Papers)*. Baltimore, MD: Association for Computational Linguistics, pp. 238–247.

Batchkarov, Miroslav et al. (2016). "A Critique of Word Similarity as a Method for Evaluating Distributional Semantic Models". In: *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*. Berlin, Germany: Association for Computational Linguistics, pp. 7–12.

Bommasani, Rishi, Kelly Davis, and Claire Cardie (2020). "Interpreting Pretrained Contextualized Representations via Reductions to Static Embeddings". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by Dan Jurafsky et al. Online: Association for Computational Linguistics, pp. 4758–4781.

Devlin, Jacob et al. (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)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, MN: Association for Computational Linguistics, pp. 4171–4186.

Firth, J. R. (1957). "A Synopsis of Linguistic Theory, 1930–1955". In: *Studies in Linguistic Analysis*. Oxford, UK: Basil Blackwell, pp. 1–32.

Frege, Gottlob (1980). *The Foundations of Arithmetic: A Logico-Mathematical Enquiry Into the Concept of Number*. 2nd rev. ed. Evanston, Ill: Northwestern University Press.

Gamallo, Pablo (2020). "CitiusNLP at SemEval-2020 Task 3: Comparing Two Approaches for Word Vector Contextualization". In: *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. Ed. by Aurelie Herbelot et al. Barcelona (online): International Committee for Computational Linguistics, pp. 275–280.

Gupta, Prakhar, Matteo Pagliardini, and Martin Jaggi (2019). *Better Word Embeddings by Disentangling Contextual N-Gram Information*.

Harris, Zellig S. (1954). "Distributional Structure". In: *WORD* 10.2-3, pp. 146–162.

Hettiarachchi, Hansi and Tharindu Ranasinghe (2021). *BRUMS at SemEval-2020 Task 3: Contextualised Embeddings for Predicting the (Graded) Effect of Context in Word Similarity*. URL: `http://arxiv.org/abs/2010.06269`.

Hill, Felix, Roi Reichart, and Anna Korhonen (2015). "SimLex-999: Evaluating Semantic Models With (Genuine) Similarity Estimation". In: *Computational Linguistics* 41.4, pp. 665–695.

Kintsch, Walter (2001). "Predication". In: *Cognitive Science* 25.2, pp. 173–202.

Liu, Qi, Matt J. Kusner, and Phil Blunsom (2020). *A Survey on Contextual Embeddings*. URL: `http://arxiv.org/abs/2003.07278`.

Mikolov, Tomáš, Kai Chen, et al. (2013). *Efficient Estimation of Word Representations in Vector Space*.

Mikolov, Tomáš, Ilya Sutskever, et al. (2013). "Distributed Representations of Words and Phrases and Their Compositionality". In: *Advances in Neural Information Processing Systems*. Ed. by C. J. Burges et al. Vol. 26. Curran Associates, Inc.

Mitchell, Jeff and Mirella Lapata (2008). "Vector-Based Models of Semantic Composition". In: *Proceedings of ACL-08: HLT*. Columbus, Ohio: Association for Computational Linguistics, pp. 236–244.
Padó, Sebastian and Mirella Lapata (2003). "Constructing Semantic Space Models from Parsed Corpora". In: *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan: Association for Computational Linguistics, pp. 128–135.
Pennington, Jeffrey, Richard Socher, and Christopher D. Manning (2014). "GloVe: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Alessandro Moschitti, Bo Pang, and Walter Daelemans. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543.
Pessutto, Lucas R. C. et al. (2020). *BabelEnconding at SemEval-2020 Task 3: Contextual Similarity as a Combination of Multilingualism and Language Models*. URL: `http://arxiv.org/abs/2008.08439`.
Vaswani, Ashish et al. (2017). "Attention Is All You Need". In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc.
Wolf, Thomas et al. (2020). *HuggingFace's Transformers: State-of-the-art Natural Language Processing*.

8 Appendix

8.1 Cost-benefit analysis of contextual embeddings

8.2 Language-specificity of window-size effects

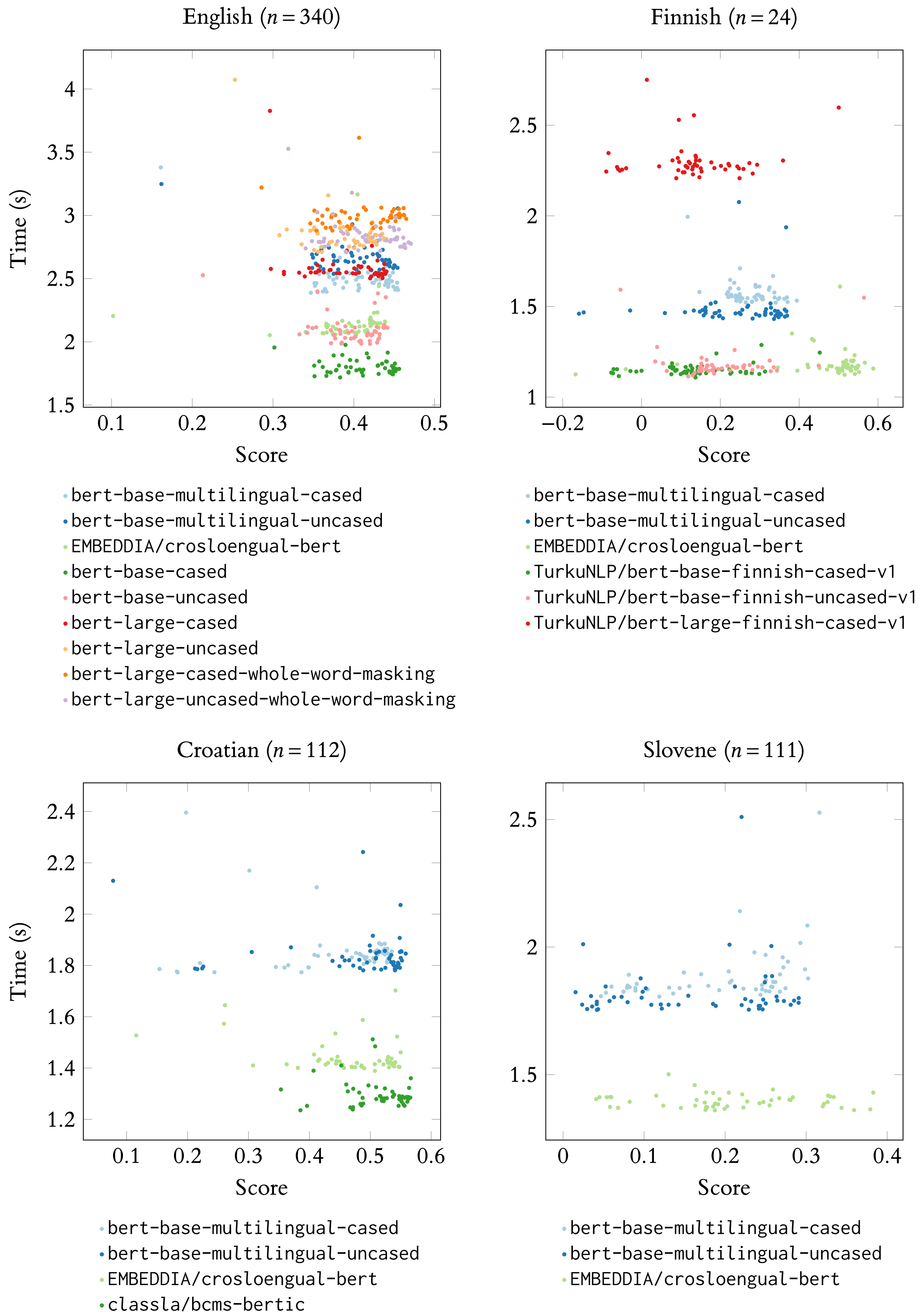

Figure 3: The time against score for static-embedding models with the additive composition operation.

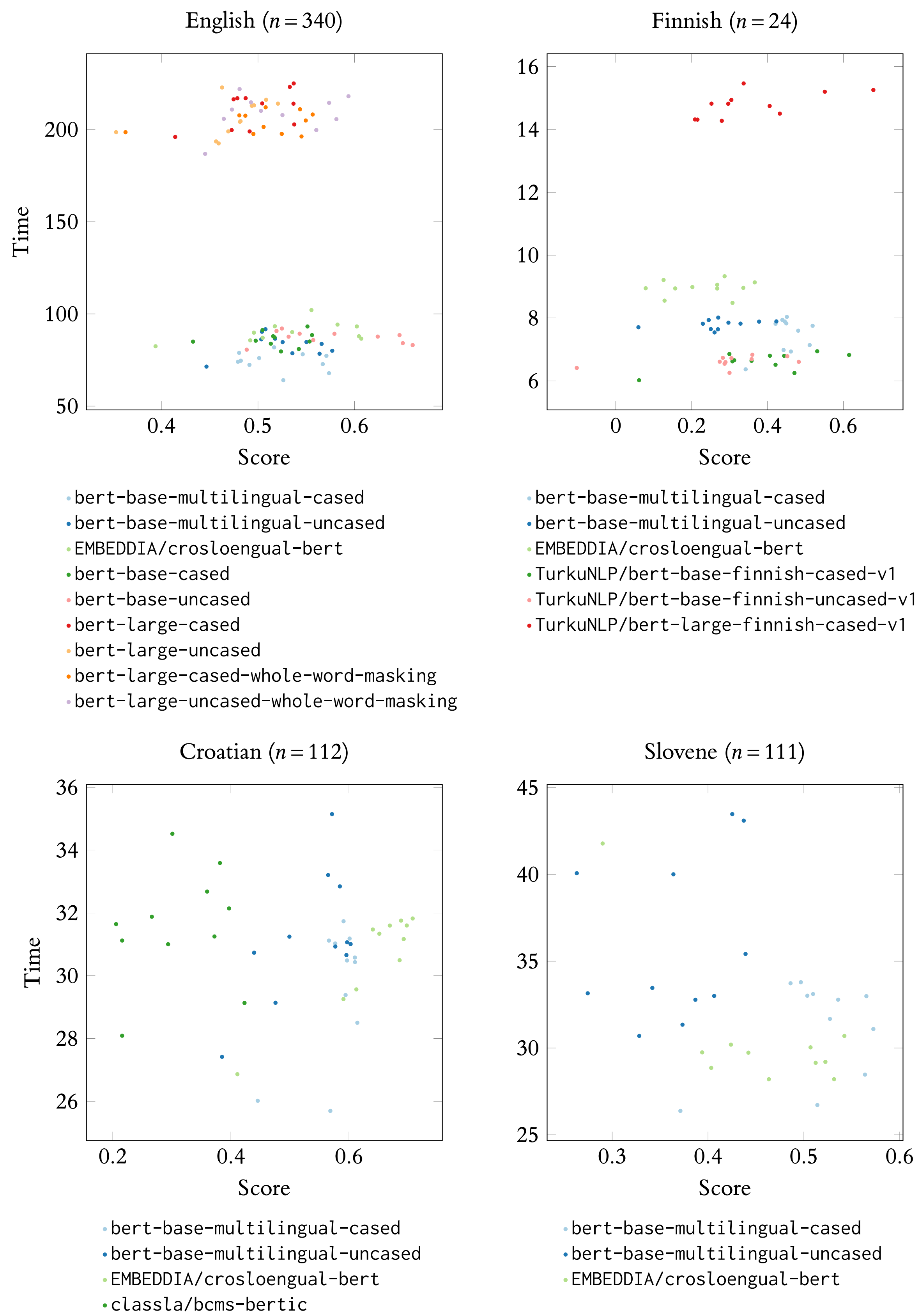

Figure 4: The time against score for contextual-embedding models with the additive composition operation.

Model name	Window size	Score
`EMBEDDIA/crosloengual-bert`	17	0.439
`bert-base-cased`	18	0.456
`bert-base-multilingual-cased`	18	0.455
`bert-base-multilingual-uncased`	19	0.455
`bert-base-uncased`	14	0.442
`bert-large-cased`	18	0.441
`bert-large-cased-whole-word-masking`	18	0.465
`bert-large-uncased`	16	0.440
`bert-large-uncased-whole-word-masking`	16	<u>0.471</u>

(a) English

Model name	Window size	Score
`EMBEDDIA/crosloengual-bert`	21	<u>0.588</u>
`TurkuNLP/bert-base-finnish-cased-v1`	0	0.452
`TurkuNLP/bert-base-finnish-uncased-v1`	0	0.564
`TurkuNLP/bert-large-finnish-cased-v1`	0	0.500
`bert-base-multilingual-cased`	3	0.394
`bert-base-multilingual-uncased`	29	0.369

(b) Finnish

Model name	Window size	Score
`EMBEDDIA/crosloengual-bert`	31	0.550
`bert-base-multilingual-cased`	32	0.535
`bert-base-multilingual-uncased`	31	0.558
`classla/bcms-bertic`	31	<u>0.567</u>

(c) Croatian

Model name	Window size	Score
`EMBEDDIA/crosloengual-bert`	11	<u>0.383</u>
`bert-base-multilingual-cased`	8	0.316
`bert-base-multilingual-uncased`	10	0.291

(d) Slovene

Figure 5: The window size that maximises the score for static-embedding models with the additive composition operation. The best score for each language is underlined.

Model name	Window size	Score
EMBEDDIA/crosloengual-bert	3	0.607
bert-base-cased	1	0.556
bert-base-multilingual-cased	1	0.574
bert-base-multilingual-uncased	3	0.577
bert-base-uncased	1	<u>0.660</u>
bert-large-cased	3	0.538
bert-large-cased-whole-word-masking	2	0.557
bert-large-uncased	1	0.521
bert-large-uncased-whole-word-masking	1	0.594

(a) English

Model name	Window size	Score
EMBEDDIA/crosloengual-bert	10	0.366
TurkuNLP/bert-base-finnish-cased-v1	1	0.615
TurkuNLP/bert-base-finnish-uncased-v1	1	0.483
TurkuNLP/bert-large-finnish-cased-v1	1	<u>0.679</u>
bert-base-multilingual-cased	3	0.519
bert-base-multilingual-uncased	2	0.423

(b) Finnish

Model name	Window size	Score
EMBEDDIA/crosloengual-bert	3	<u>0.708</u>
bert-base-multilingual-cased	5	0.614
bert-base-multilingual-uncased	6	0.603
classla/bcms-bertic	10	0.423

(c) Croatian

Model name	Window size	Score
EMBEDDIA/crosloengual-bert	3	0.542
bert-base-multilingual-cased	3	<u>0.572</u>
bert-base-multilingual-uncased	9	0.439

(d) Slovene

Figure 6: The window size that maximises the score for contextual-embedding models with the additive composition operation. The best score for each language is underlined.

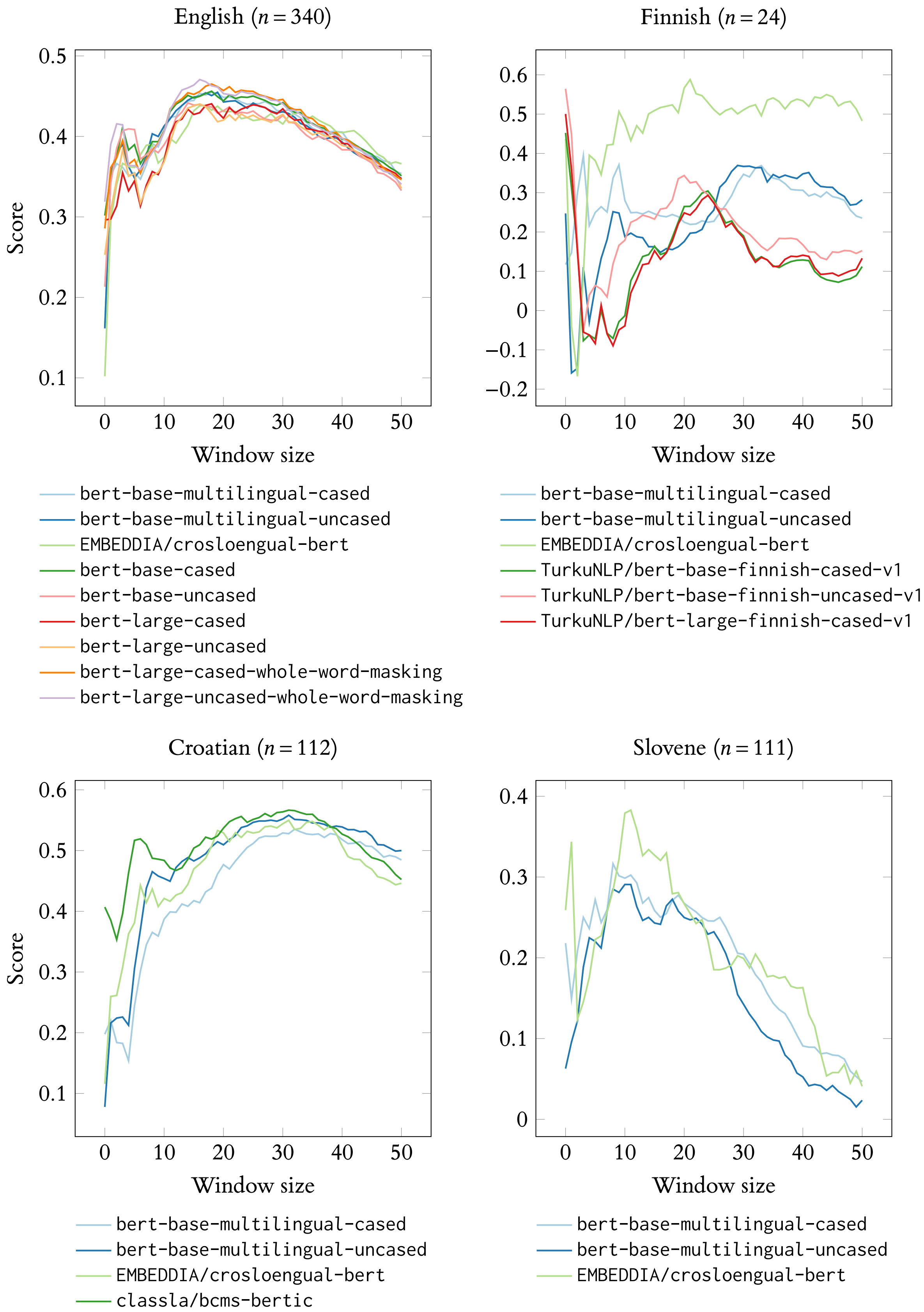

Figure 7: The score against window size for static-embedding models with the additive composition operation.

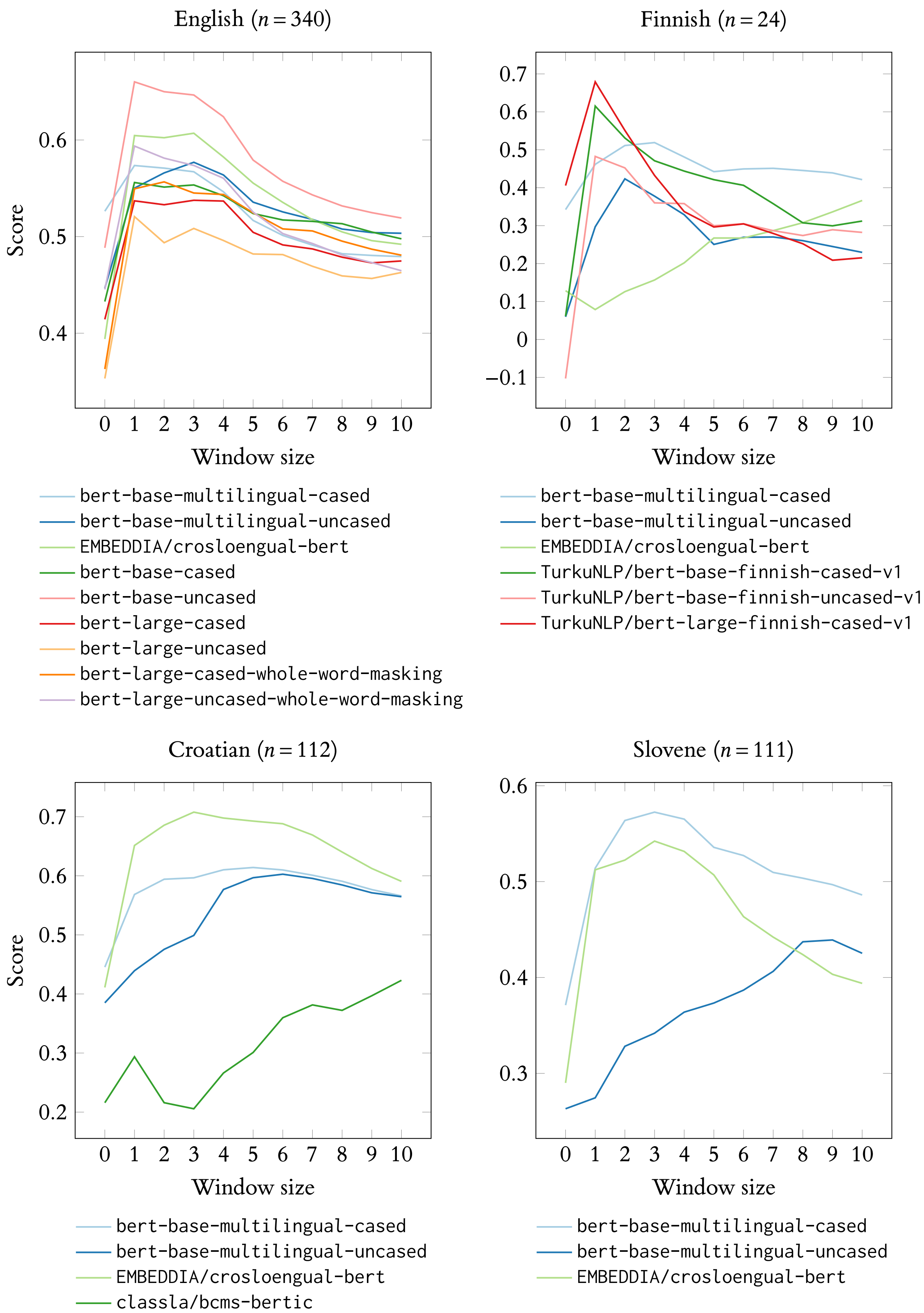

Figure 8: The score against window size for contextual-embedding models with the additive composition operation.