XL-DURel: Finetuning Sentence Transformers for Ordinal Word-in-Context Classification

Sachin Yadav, Dominik Schlechtweg
Institute for Natural Language Processing, University of Stuttgart
sachindv42@gmail.com, schlecdk@ims.uni-stuttgart.de

Abstract

We propose XL-DURel, a finetuned, multilingual Sentence Transformer model optimized for ordinal Word-in-Context classification. We test several loss functions for regression and ranking tasks managing to outperform previous models on ordinal and binary data with a ranking objective based on angular distance in complex space. We further show that binary WiC can be treated as a special case of ordinal WiC and that optimizing models for the general ordinal task improves performance on the more specific binary task. This paves the way for a unified treatment of WiC modeling across different task formulations.

1 Introduction

The Ordinal Graded Word-in-Context (OGWiC) task asks to predict the level of semantic proximity between two word usages on an ordinal scale (Schlechtweg et al., 2025). While it builds on the earlier WiC (Pilehvar and Camacho-Collados, 2019) and GWiC (Armendariz et al., 2020) tasks, it can be distinguished from these by being formulated as an **ordinal classification** task (Sakai, 2021). This is similar to ranking in that labels are inherently ranked, but also similar to classification in that exact labels have to be predicted for each test instance, instead of merely an ordering of instances as in ranking tasks. State-of-the-art OGWiC models employ pre-trained Language Models like XLM-R (Conneau et al., 2019) to generate contextualized embeddings for the target word in two different contexts and finetune these with loss functions tailored for **binary** or **nominal** data such as the contrastive or the cross-entropy loss (e.g. Cassotti et al., 2023; Kuklin and Arefyev, 2025). We conjecture that these models do not sufficiently exploit the ranking signal provided by OGWiC training data. In this study, we aim to overcome this limitation by employing loss functions directly optimizing ranking or regression objectives. We compare these against previous models trained with binary classification objectives and manage to outperform the latter on ordinal and binary data with a ranking objective based on angular distance in complex space. By improving performance on the binary and the ordinal formulation of the task through the same model, we pave the way for a unified treatment of WiC modeling. We publish our top-performing model, **XL-DURel**, which can be employed as highly optimized, fine-grained and multilingual contextualized embedder for word-meaning-related tasks.[1]

2 Related Work

2.1 WiC Task

The challenge of capturing the dynamic semantics of words has led to the development of various evaluation benchmarks. One notable contribution in this area is the Word-in-Context (WiC) task and the corresponding dataset, introduced by Pilehvar and Camacho-Collados (2019). The WiC task is designed to assess context-sensitive word representations by framing it as a binary classification problem. In this task, each instance consists of a target word w, and two usages (or sentences) u_1 and u_2. The objective is to determine whether the meaning of the target word remains consistent across the two usages. If the meaning is the same, the instance is labeled 'TRUE' (or '1') as in pair (1,2):

(1) The expanded **window** will give us time to catch the thieves.

(2) You have a two-hour **window** of clear weather to finish working on the lawn.

If the meaning differs, it is labeled 'FALSE' (or '0') as in pair (3,4):

[1] https://huggingface.co/sachinn1/xl-durel

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 338–351
December 20-24, 2025 ©2025 Association for Computational Linguistics

(3) There's a lot of trash on the **bed** of the river.

(4) I keep a glass of water next to my **bed** when I sleep.

Performance on this task is usually evaluated with Accuracy. The first WiC dataset by Pilehvar and Camacho-Collados (2019) was constructed from sense-annotated lexical resources such as WordNet (Fellbaum, 2005), VerbNet (Schuler, 2005) and Wiktionary[2].

2.2 GWiC Task

The Graded Word Similarity in Context (GWiC) task (Armendariz et al., 2020), introduced as part of SemEval-2020 Task 3, aims to evaluate how well computational models can capture graded word similarity in different contexts such as (5) and (6):

(5) ...These young **men** displayed true Rajput chivalry. Akbar was so impressed with the bravery of these two **warriors** that he commissioned...

(6) ...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.

In one of the subtasks, participants were tasked with predicting the absolute similarity rating for each word pair within each context on a scale from 0 to 10. For word pair *man* and *warrior*, the gold similarity score is 7.88 in (5) and 3.27 in (6). The shared task used the Harmonic Mean of Pearson and Spearman correlations as an evaluation metric. It can thus be interpreted as a mixture of a regression and a ranking task.

2.3 OGWiC Task

The Ordinal Graded Word-in-Context (OGWiC) task was introduced as part of the CoMeDi shared task (Schlechtweg et al., 2025), focusing on nuanced and interpretable evaluation of word meaning in context. It aims to address the problems of the WiC and GWiC tasks by defining an ordinal classification task requiring participants to exactly reproduce the median annotated label for a word usage pair on a scale from 1 (unrelated) to 4 (closely related).[3] For example, the pair (7,8) receives label 4 (identical) while pair (7,9) receives the lower label 2 (distantly related):

(7) ...the dismissal last month of the commandant and two other generals of the provincial police, reportedly for **graft**.

(8) We try to live with lies and corruption and fraud and **graft** and violence and exploitation and...

(9) The second, which is spread while warm on strips of coarse cotton, or strong paper, and wrapped directly about the **graft**, answering at once to tie and to protect it, is composed of equal parts of bees-wax, turpentine, and resin.

OGWiC is similar to the previous WiC and GWiC tasks, but limits the label set in predictions and penalizes stronger deviations from the true label. This makes OGWiC an **ordinal classification task** (Sakai, 2021), in contrast to binary classification (WiC) or ranking (GWiC). Predictions are evaluated against the median labels with the ordinal version of Krippendorff's α (Krippendorff, 2018).

Two models excelled in the CoMeDi shared task (Choppa et al., 2025; Kuklin and Arefyev, 2025): **XL-LEXEME** (Cassotti et al., 2023) builds upon the Sentence Transformers architecture (Reimers and Gurevych, 2019) and employs a bi-encoder framework within a Siamese network. Vectors are initialized with XLM-RoBERTa (XLM-R, Conneau et al., 2019) and their similarity is directly optimized using a contrastive loss function (Hadsell et al., 2006), which minimizes the distance between embeddings of sentences with the same meaning (label 'TRUE') while maximizing the distance between embeddings of sentences with different meanings (label 'FALSE') around a preselected margin. At test time, the model predicts the similarity between two usages using the finetuned base model and thresholds it to infer ordinal labels (see Section 4.4). A similar approach is taken by the **DeepMistake** model (Arefyev et al., 2021). Vectors are initialized with XLM-R, sentences concatenated and jointly encoded. Then the target word vectors are extracted and jointly fed into a binary classification head. The model is finetuned using the cross-entropy loss. Similar to XL-LEXEME, DeepMistake is trained on binary WiC-like data. At test time, the model predicts the probability of label 'TRUE' and thresholds it to infer ordinal labels.

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

[3] Find more details on the scale in Appendix A.

Dataset	Train	Dev	Test	Cosine	Binary	Ordinal
CoMeDi	47,833	8,287	15,332	$4 \rightarrow 1.0$ $3 \rightarrow \frac{2}{3}$ $2 \rightarrow \frac{1}{3}$ $1 \rightarrow 0.0$	$4 \rightarrow 1$ $3 \rightarrow 1$ $2 \rightarrow 0$ $1 \rightarrow 0$	
WiC	251,972	8,381	6,400	$1 \rightarrow 1.0$ $0 \rightarrow \frac{1}{3}$		$1 \rightarrow 4$ $0 \rightarrow 2$
WiC+CoMeDi	299,805	16,668	21,732	as above	as above	as above

Table 1: Dataset statistics and label mappings.

3 Data

3.1 Ordinal WiC

We use the OGWiC data provided by the CoMeDi shared task organizers (Schlechtweg et al., 2025), available in starting kit 1.[4] The data comprises 71k word usage pairs sampled from ordinal WiC datasets across multiple languages, including Chinese, English, German, Norwegian, Russian, Spanish and Swedish. (See Table 3 in Appendix B for details.) The data was cleaned in various steps: Initially, instances with fewer than two annotations or those marked with any "Cannot decide" were excluded. Instances with significant annotator disagreement (more than one point on the scale) were also removed. A median judgment was calculated for each instance, retaining only integer medians for task consistency. The data was split by language, with 70% allocated to training, 20% to testing, and 10% to development, ensuring that no target word overlapped between these splits. (See Table 4 in Appendix B for details.)

3.2 Binary WiC

In addition to the CoMeDi data, our study incorporates the datasets used for training the XL-LEXEME model (see Section 2.3), which combines three established multilingual benchmarks: `XL-WiC` (Raganato et al., 2020), `MCL-WiC` (Martelli et al., 2021), and `AM2iCo` (Liu et al., 2021). These benchmarks are widely used for evaluating word meaning in context.

XL-WiC is a multilingual extension of the original WiC dataset (Pilehvar and Camacho-Collados, 2019), containing over 112k sentence pairs across 12 languages: Bulgarian, Chinese, Croatian, Danish, Dutch, Estonian, Farsi, French, German, Italian, Japanese, and Korean. Training data is available for German, French, and Italian while development and test sets are provided for all 12 languages. Most of the data was automatically extracted from WordNet or Wiktionary sense inventories without direct human annotation. The dataset is distributed together with the original English WiC dataset comprising roughly 7K sentence pairs.

MCL-WiC (Multilingual and Cross-lingual Word-in-Context Disambiguation) comprises approximately 10k sentence pairs spanning five languages: Arabic, Chinese, English, French, and Russian. The dataset contains data for two distinct subtasks: (i) multilingual WiC classification within individual languages, and (ii) cross-lingual classification comparing sentences from different languages. The dataset is specifically designed to evaluate model performance across both high- and medium-resource language settings. Unlike XL-WiC, which relies on sense inventories, MCL-WiC is entirely human-annotated.

AM2iCo (Adversarial and Multilingual Meaning in Context) contains roughly 196k instances spanning 14 language pairs and 15 typologically diverse languages, including English, German, Russian, Japanese, Korean, Mandarin Chinese, Arabic, Indonesian, Finnish, Turkish, Basque, Georgian, Urdu, Bengali, and Kazakh. The dataset supports evaluation of word meaning in context both within individual languages and across different languages, with a particular focus on low-resource scenarios. AM2iCo is constructed by automatically extracting WiC pairs from Wikipedia, and then filtering them through human validation and adversarial filtering.[5]

Cassotti et al. constructed the training set for XL-LEXEME by merging the official training splits from the three above-described datasets. To further

[4] `https://comedinlp.github.io/`

[5] Adversarial filtering is a strategy to make a dataset harder and more useful by removing easy examples that models can solve without actually understanding the task.

augment the training data, they randomly sampled 75% of each dataset's development data and added it to the training pool. The remaining 25% of the development data was reserved for hyperparameter tuning and validation.

As part of our study, we concatenate the CoMeDi shared task dataset and the XL-LEXEME dataset into a unified resource. For clarity, we refer to the XL-LEXEME dataset concatenation henceforth as "WiC (train/dev)". We further refer to the CoMeDi shared task data as "CoMeDi (train/dev/test)". Additionally, we include the original WiC and MCL-WiC test datasets for evaluation in our experiments. We refer to these as "WiC (test)". Statistics for the final datasets are given in Table 1.[6]

3.3 Label Mapping

As summarized in Table 1, we apply a systematic label mapping procedure to align the datasets for unified model training and evaluation. Specifically, we transform binary and ordinal labels to cosine-like labels (interval $[0, 1]$) if needed for the respective loss function used for training (see Section 4.2). Similarly, we transform ordinal labels to binary labels if needed. As summarized in Table 1, for ordinal-to-cosine mapping, we utilize `Min-Max-Scaling` to map labels to the interval $[0, 1]$. This maps the ordinal labels as follows: 1 $\rightarrow$ 0.0, 2 $\rightarrow \frac{1}{3}$, 3 $\rightarrow \frac{2}{3}$, and 4 $\rightarrow$ 1.0. For binary-to-cosine mapping, we map label 1 (same sense) to cosine label 1.0 to align with annotation level 4 (identical) on the ordinal scale (cf. Table 2 in Appendix A). Binary label 0 (different sense) is mapped to cosine label $\frac{1}{3}$ to align with level 2 (polysemy) on the ordinal scale, based on the assumption that most pairs of usages, especially from the same target word, will be semantically related, e.g. by contiguity or similarity. For ordinal-to-binary mapping, we group ordinal labels 1 and 2 as binary label 0, and labels 3 and 4 as binary label 1, which is motivated by the idea that ordinal label 2 (polysemy) is a relation *between* senses while ordinal label 3 (context variance) is a variation *within* a sense (see Appendix A). This mapping is needed in some cases for evaluation. Following the same logic as for binary-to-cosine, for binary-to-ordinal mapping we assign binary label 1 to ordinal label 4, and binary label 0 to ordinal label 2.

[6]Unintentionally, we skipped 10 files in Cassotti et al.'s training data package, i.e., AM2iCo dev ar-en/bn-en, XL-WiC dev bg-bg/da-da/en-en/et-et/fr-fr/zh-zh, WiC dev en-en, MCL-WiC dev en-en. In total, these are 12,512 instances.

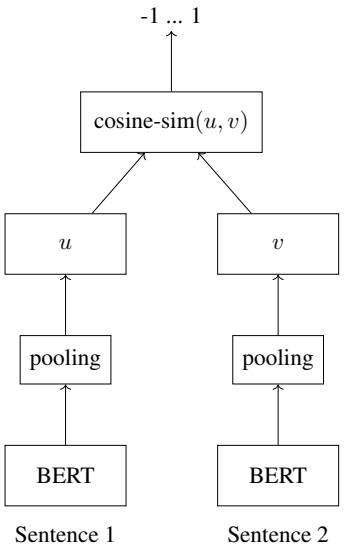

Figure 1: SBERT architecture at inference.

4 Model

We employ Sentence-BERT (SBERT, Reimers and Gurevych, 2019), a modification of BERT designed to generate semantically meaningful sentence embeddings for efficient semantic similarity comparison. Unlike standard BERT, which requires joint processing of sentence pairs, SBERT uses a *siamese* or *triplet* network architecture. Each sentence is independently encoded by a BERT-based model with shared weights and pooled, resulting in fixed-size sentence embeddings. SBERT is implemented using the Sentence Transformers library (Reimers and Gurevych, 2020), which offers a flexible interface for finetuning SBERT models with various loss functions. Most of these aim to adjust base model parameters so that similarities between sentence embeddings align with gold similarity values. A standard choice for the optimized similarity metric is the Cosine Similarity or its inverse, the Cosine Distance (Salton and McGill, 1983). See Figure 1 for an illustration of the SBERT architecture.

4.1 Target Word Marking

The WiC task requires semantic disambiguation at the token level rather than at the sentence level. This presents a challenge for SBERT, which is primarily designed for sentence-level embedding and comparison. We adopt XL-LEXEME's (see Section 2.3) strategy to adapt sentence embeddings to focus on specific target words within their contexts by marking the target word in each usage u with

special tokens:

$$u = w_1, \ldots, \langle t \rangle, w_{t_i}, \ldots, w_{t_i+k}, \langle /t \rangle, \ldots, w_N$$

where $\langle t \rangle$ and $\langle /t \rangle$ denote the opening and closing markers for the target word w_t, and w_i represents individual words in the sentence. Inputs are truncated to a maximum sequence length of 128 tokens. After truncation, we additionally append the `[CLS]` and the `[SEP]` token before and after the input sequence, respectively.

4.2 Loss Functions

We experiment with the following loss functions to optimize model performance. If not stated differently, we use the cosine similarity/distance and the loss is calculated as mean per batch.

Contrastive Loss expects two embeddings (u, v) and a binary label $y \in \{0, 1\}$ as inputs (Hadsell et al., 2006). It drives the similarity between positive pairs towards 1 and that between negative pairs to decrease to a margin. In the Sentence Transformers library, the loss is defined as:

$$\mathcal{L} = \frac{1}{2}\Big(y \cdot d(u,v)^2 + (1-y) \cdot \max(0, m - d(u,v))^2\Big)$$

where

- $d(u, v)$ is the distance between the embeddings and
- m is the margin hyperparameter, which specifies the minimum required distance between dissimilar pairs.

Selecting the optimal margin for different datasets may be challenging (Huang et al., 2024). Also, relative distances with ordered label sets with more than two classes cannot be encoded making the loss ill-suited for ranking tasks.

Cosine Similarity Loss expects two embeddings (u, v) and a continuous similarity label $y \in [0, 1]$ as inputs. It is defined as the mean squared difference between the predicted similarities between embeddings and the ground truth label:

$$\mathcal{L} = \|\cos(u, v) - y\|_2$$

where

- $\|\cdot\|_2$ is the L2 norm.

The mean squared error is a common loss function used in regression tasks. However, according to Huang et al. (2024), it is unsuitable for classification tasks because noise does not follow a normal distribution (cf. Ciampiconi et al., 2024).

CoSENT Loss expects two embeddings (u, v) and a continuous similarity label $y \in [0, 1]$ as inputs (Huang et al., 2024). It trains the embeddings so that the higher the similarity label between pairs, the higher the similarity of their embeddings:

$$\mathcal{L} = \log\left(1 + \sum_{y(u,v) > y(k,l)} \exp(\lambda(s(k,l) - s(u,v)))\right)$$

where

- $s(u, v)$ is the similarity between the embeddings,
- $y(u, v) > y(k, l)$ defines the set of embedding pairs (k, l) for which the ground truth label $y(k, l)$ is smaller than $y(u, v)$ and
- λ is a hyperparameter for amplification.[7]

The loss is computed as sum over all pairs (u, v) in the batch. In contrast to the contrastive loss operating *within* the sentence pairs, CoSENT focuses on maintaining ranking consistency *between* the learned similarity of sentence pairs within the entire set and their similarity labels (Huang et al., 2024). This also distinguishes it from the Cosine Similarity Loss which operates only on individual pairs and only implicitly optimizes ranking consistency.

AnglE Loss expects two embeddings (u, v) and a continuous similarity label $y \in [0, 1]$ as inputs. It uses the CoSENT Loss function (see above) with a different similarity measure, i.e., the angle difference in complex space (Li and Li, 2023). The AnglE Loss was introduced to address a key limitation of the cosine function: The gradient of the cosine function tends to approach zero as it nears its maximum or minimum values, which can hinder the optimization process. According to Li and Li, this is not the case for the angle difference in complex space.

4.3 Optimization

During training, the parameters of the base model are adjusted in order to minimize the respective

[7] The Sentence Transformers library makes two specific implementation choices: (i) Pairs (k, l) that do not meet the condition $y(u, v) > y(k, l)$ are masked by subtracting a large constant (i.e., 10^{12}) from their score difference, making their contribution negligible in the exponential term. (ii) For numerical stability, a zero is appended to the set of all cosine similarities in a batch where $y(u, v) > y(k, l)$ to guarantee numerical stability in cases where the set $y(u, v) > y(k, l)$ is empty.

loss function from Section 4.2. For all experiments, we keep the following settings constant: We use XLM-R-large as our base model and optimize with `AdamW`. We set the learning rate to 1×10^{-5}, the batch size to 32 and use no weight decay (0.0). All other settings are kept at their default values. For Contrastive Loss we set the margin $m = 0.5$, and for the ranking losses (CoSENT, AnglE Loss) we use the default $\lambda = 20$. We train for 10 epochs, with a linear warm-up over 10% of the total training steps. We evaluate at every 25% of an epoch via Average Precision[8] (Contrastive Loss) or Spearman correlation (rest) between cosine similarities and gold labels on dev data (see Section 5).[9] The final checkpoint is chosen by highest performance on dev data.

The base model, XLM-R-large, contains 561M parameters. All experiments are conducted on a Linux-based server running Fedora 42, equipped with NVIDIA RTX A6000 GPUs (48 GB VRAM per GPU) and dual Intel Xeon CPUs. We utilize a single GPU per run and estimate the computational runtime per model run to be approximately 40–50 GPU hours.[10]

4.4 Thresholding

For all models, we adopt the CoMeDi shared task baseline approach to map cosine similarities to ordinal labels. At test time, similarities are mapped to ordinal labels using three thresholds θ, which are optimized on the dev set by minimizing the following loss function (cf. Choppa et al., 2025):

$$\mathcal{L} = 1 - \alpha(\mathbf{y}, \hat{\mathbf{y}}_\theta)$$

where

- $\mathbf{y}$ are gold labels,
- $\hat{\mathbf{y}}_\theta$ are predicted labels according to thresholds θ on similarity predictions $\hat{\mathbf{y}}$ and
- α is Krippendorff's α.

Krippendorff's α is the task evaluation metric (see Section 5). We aim to find optimal values for θ. This threshold optimization is performed on the dev data and separately for each language to account for language-specific distributional differences in similarity scores. It uses the Nelder–Mead simplex method (Nelder and Mead, 1965). (Find induced thresholds in Appendix C.)

4.5 Baseline Models

In our experiments, we employ a number of simple baseline models, as described below. All models use thresholding as explained in Section 4.4 for mapping similarities to ordinal labels.

SBERT uses the cosine similarity on a non-finetuned SBERT model initialized with XLM-R-Large.

XL-LEXEME uses XLM-R-Large as base model and was finetuned with SBERT using the Contrastive Loss on WiC train and dev (see Sections 2.3 and 3).

XL-LEXEME CoMeDi is the XL-LEXEME result reported in the CoMeDi shared task. Notably, it achieved the second-best performance. We use this as a baseline in our evaluation.

DeepMistake CoMeDi is the DeepMistake result reported in the CoMeDi shared task. It achieved the best performance. It uses XLM-R-Large as base model and was finetuned with the cross-entropy loss for binary classification on MCL-WiC train and dev, the Spanish subset of XL-WSD (Pasini et al., 2021) and a binarized version of the Spanish DWUG dataset (Zamora-Reina et al., 2022) (see Sections 2.3 and 3). Because Spanish DWUG is part of our test data, we report additional average performance without Spanish in Section 6.

5 Evaluation

Following the CoMeDi shared task, we use ordinal **Krippendorff's** α (Krippendorff, 2018) as evaluation measure for ordinal classification. It penalizes stronger deviations from the gold label more heavily. It has the additional advantage of controlling for expected disagreement and has been demonstrated to be superior to other measures such as Mean Absolute Error for ordinal classification (Sakai, 2021). We also use Spearman's rank correlation coefficient (ρ) between continuous similarities and gold ordinal labels to assess the rank alignment of model predictions. This enables us to evaluate performance without inducing thresholds. We further apply the nominal version of Krippendorff and the Spearman correlation for binary label

[8] We also tried Spearman correlation and did not observe a considerable difference in results.

[9] We also experimented with using the angle difference on dev for AnglE Loss, but did not outperform the cosine similarity.

[10] GitHub Copilot was used to assist with coding during the implementation.

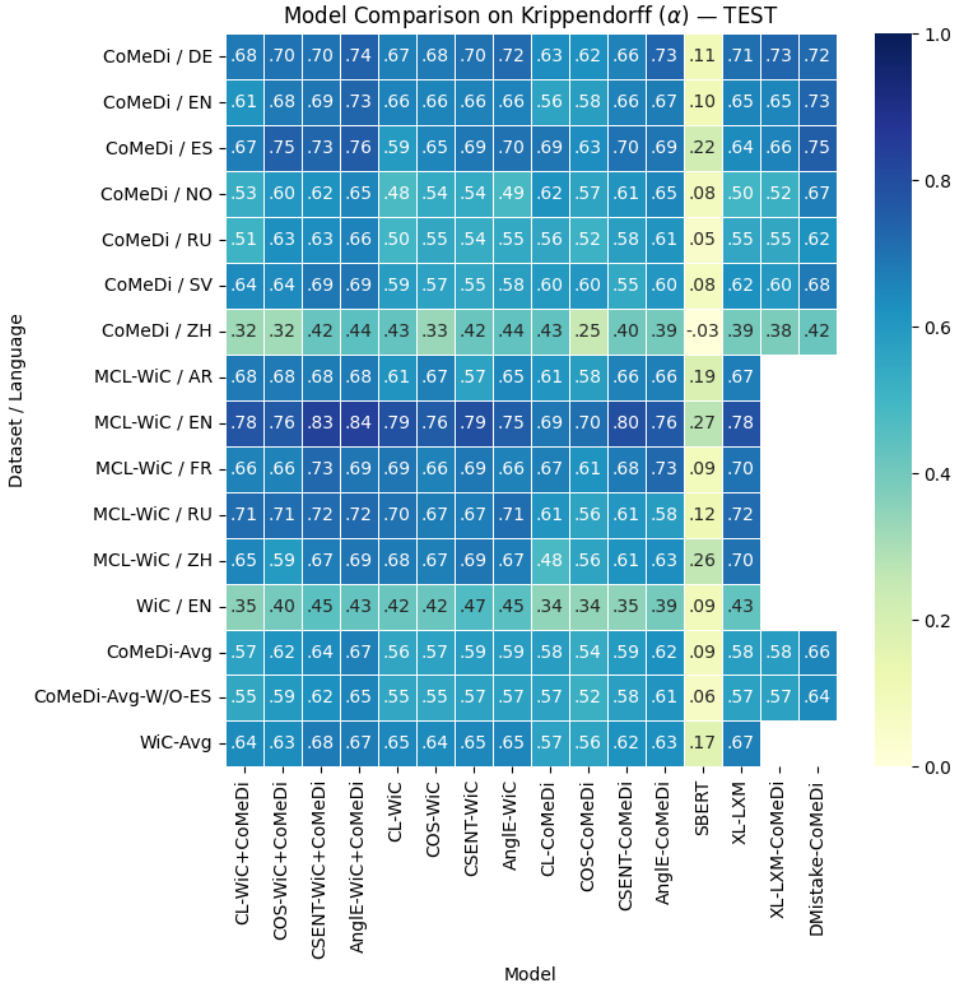

Figure 2: Model evaluation with Krippendorff's α on binary and ordinal test data. CL = Contrastive Loss, COS = Cosine Similarity Loss, CSENT = CoSENT Loss, AnglE = AnglE Loss, XL-LXM = XL-LEXEME, DMistake = DeepMistake.

evaluation. During training, similarities are compared to ordinal labels with Spearman correlation and to binary labels with the Average Precision.

6 Experiments

We now test which loss functions (see Section 4.2) and data combinations (see Section 3) improve performance on the ordinal CoMeDi test data over the baselines described in Section 4.5. We additionally report performance on binary WiC data to understand whether optimization for ordinal data hurts the binary task. As finetuning is computationally expensive, we perform one run for each model.[11] Models are evaluated with Krippendorff's α as described in Section 5 based on their cosine predictions binned to ordinal labels (see Section 4.4).[12] All experiments follow the training setup described in Section 4.3. Results are shown in Figure 2.[13]

Loss function First of all, we see that finetuning has a strong effect on performance. For this, compare all models against SBERT, which is the only non-finetuned model. With a performance of .67, AnglE Loss achieves the best result on CoMeDi test data ('CoMeDi-Avg') when using both WiC and CoMeDi training data (AnglE-WiC+CoMeDi). With performances of .64 and .62 respectively, it is followed by the CoSENT Loss (CSENT-WiC+CoMeDi) and the Cosine Similar-

[11] We selectively re-ran models and observed variation of average performances between ±.01–.03.

[12] We also experimented with using the angle difference on test for AnglE Loss, but did not outperform the cosine similarity.

[13] In Appendix D, we give an additional performance evaluation on the raw cosine predictions with Spearman rank correlation.

ity Loss (COS-WiC+CoMeDi). All these models clearly outperform the published version of XL-LEXEME (XL-LXM), its retrained model version (CL-WiC) and the retrained version with additional ordinal data (CL-WiC+CoMeDi), which have performances of .58, .56 and .57, respectively. The top model AnglE-WiC+CoMeDi outperforms the latter ones on all languages and on average by a large margin. The same holds for the published XL-LEXEME result from the CoMeDi shared task (XL-LXM-CoMeDi), which is outperformed with .67 vs. .58. It further slightly outperforms the shared task winning model DeepMistake (.67 vs. .66). This is also the case if we exclude Spanish ('CoMeDi-Avg-W/O-ES'), which was reported additionally by the task organizers because DeepMistake was trained on part of the test data for Spanish. Notably, we reach this performance by optimizing one unified model while the DeepMistake result was obtained by optimizing multiple models tailored to specific languages.[14] On binary WiC ('WiC-Avg'), top performance is reached by the CoSENT Loss model relying on both binary and ordinal training data (CSENT-WiC+CoMeDi) with .68, closely followed by the AnglE Loss relying on the same data (AnglE-WiC+CoMeDi) and XL-LEXEME with .67, respectively. Hence, on the ordinal and the binary task ranking losses show top performance, where the advantage to the classification loss is more pronounced for the ordinal task. While the regression loss is not competitive for either task, it shows a clear advantage over the classification loss on the ordinal task. These results are in line with the motivations given for the loss functions in Section 4.2: AnglE and CoSENT Loss are explicitly optimizing a ranking objective, which exploits the inherent ordering of ordinal labels. Further, the AnglE Loss improves optimization over the CoSENT Loss and other cosine-based losses, presumably because it avoids killed gradients occurring with the cosine similarity.

Training data Note that training on purely ordinal data yields good baseline performance across tasks, especially with the AnglE Loss (AnglE-CoMeDi). For binary data, this is also the case, but there is a larger performance difference to top models on the ordinal task (e.g. AnglE-WiC). Moreover, we clearly observe that combining ordinal and binary data improves performance on the ordinal task across all loss functions. Compare for example performances of AnglE-WiC/CoMeDi vs. AnglE-WiC+CoMeDi or CSENT-WiC/CoMeDi vs. CSENT-WiC+CoMeDi. There is a clear average performance improvement on the ordinal task. Similarly for performance on the binary task, but only for the ranking losses AnglE and CoSENT. However, for the Contrastive and the Cosine Similarity Loss the performance drop is negligible.

Ordinality Regardless of training data, the ordinal training signal turns out to be beneficial for both tasks. In order to see this, compare e.g. CL-CoMeDi vs. AnglE-CoMeDi. The former binarizes the data while the latter keeps the ordinal information. The performance difference is striking with .58 vs. .62 for the ordinal task and, interestingly, .57 vs. .63 for the binary task. This indicates that fine-grained semantic proximity information helps the model to better learn binary meaning distinctions, which is also supported by the fact that the best purely ordinal model approaches the performance of the best purely binary model on the binary task (AnglE-CoMeDi vs. CL-WiC) despite being trained on much less and out-of-distribution data.

7 Conclusion

We compared several loss functions for classification, regression and ranking to finetune OGWiC models. Our top model outperformed previous models on ordinal and binary WiC data with a ranking objective based on angular distance in complex space. Overall, we found that using the AnglE Loss can be recommended, both for the ordinal and the binary WiC task. Similarly, mixing ordinal and binary training data turned out to be beneficial for both tasks. These results suggest that binary WiC can be treated as a special case of ordinal WiC and that optimizing models for the general ordinal task improves performance on the more specific binary task. In the future, WiC task setups should try to unify these approaches in order to make use of the full power of WiC training signals from multiple types of data. Further, we should try to optimize models more directly for ordinal classification instead of ranking. Currently, our model first predicts a dense similarity which we then discretize in an independent step to ordinal labels through thresholds. However, there are also loss functions directly optimizing for ordinal labels, like Cumulative Link models (Vargas et al., 2020). We would like to

[14] We provide the finetuned AnglE-WiC+CoMeDi model under the name "XL-DURel" at https://huggingface.co/sachinn1/xl-durel. Find the code for reproducing our results at https://github.com/sachinn12/XL-DURel.

test such models for OGWiC. Moreover, similar to Loke et al. (2025), we would like to test Large Language Models such as Llama (Touvron et al., 2023). These profit from massive amounts of parameters and training data and can be directly instructed to predict an ordinal number. Note, however, that our current approach has certain advantages over this: It is theoretically motivated by employing direct ranking optimization. Also, it is small and efficient making the model applicable to large amounts of data.

Limitations

We tested our hypotheses using particular data, base models and training architectures. In future research, these should be varied to test whether they have an influence on effects. Specifically, it should be tested whether the AnglE Loss turns out to be beneficial for the ordinal and binary task with additional test data. It is also unclear why the AnglE Loss performs better with the cosine similarity than with the angle difference in complex space at test time although the latter is optimized during training.

Acknowledgments

This study is an extension of Sachin Yadav's master thesis (Yadav, 2025). We thank Lucas Möller for feedback regarding the implementation. We further thank Pierluigi Cassotti, Roksana Goworek and Haim Dubossarsky for help on reproducing XL-LEXEME results.

References

Anna Aksenova, Ekaterina Gavrishina, Elisei Rykov, and Andrey Kutuzov. 2022. RuDSI: Graph-based word sense induction dataset for Russian. In *Proceedings of TextGraphs-16: Graph-based Methods for Natural Language Processing*, pages 77–88, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Nikolay Arefyev, Maksim Fedoseev, Vitaly Protasov, Daniil Homskiy, Adis Davletov, and Alexander Panchenko. 2021. DeepMistake: Which senses are hard to distinguish for a word-in-context model. volume 2021-June, pages 16–30.

Carlos Santos Armendariz, Matthew Purver, Senja Pollak, Nikola Ljubešić, Matej Ulčar, Ivan Vulić, and Mohammad Taher Pilehvar. 2020. SemEval-2020 task 3: Graded word similarity in context. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 36–49, Barcelona (online). International Committee for Computational Linguistics.

Andreas Blank. 1997. *Prinzipien des lexikalischen Bedeutungswandels am Beispiel der romanischen Sprachen*. Niemeyer, Tübingen.

Pierluigi Cassotti, Lucia Siciliani, Marco de Gemmis, Giovanni Semeraro, and Pierpaolo Basile. 2023. Xl-lexeme: Wic pretrained model for cross-lingual lexical semantic change. In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

Jing Chen, Emmanuele Chersoni, Dominik Schlechtweg, Jelena Prokic, and Chu-Ren Huang. 2023. ChiWUG: A graph-based evaluation dataset for Chinese lexical semantic change detection. In *Proceedings of the 4th International Workshop on Computational Approaches to Historical Language Change*, Singapore. Association for Computational Linguistics.

Tejaswi Choppa, Michael Roth, and Dominik Schlechtweg. 2025. Predicting median, disagreement and noise label in ordinal Word-in-Context data. In *Proceedings of Context and Meaning: Navigating Disagreements in NLP Annotation*, pages 65–77, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Lorenzo Ciampiconi, Adam Elwood, Marco Leonardi, Ashraf Mohamed, and Alessandro Rozza. 2024. A survey and taxonomy of loss functions in machine learning. *Preprint*, arXiv:2301.05579.

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

Christiane Fellbaum. 2005. Wordnet and wordnets. encyclopedia of language and linguistics.

R. Hadsell, S. Chopra, and Y. LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742.

Anna Hätty, Dominik Schlechtweg, and Sabine Schulte im Walde. 2019. SURel: A gold standard for incorporating meaning shifts into term extraction. In *Proceedings of the 8th Joint Conference on Lexical and Computational Semantics*, pages 1–8, Minneapolis, MN, USA.

Xiang Huang, Hao Peng, Dongcheng Zou, Zhiwei Liu, Jianxin Li, Kay Liu, Jia Wu, Jianlin Su, and Philip S. Yu. 2024. Cosent: Consistent sentence embedding via similarity ranking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:2800–2813.

Klaus Krippendorff. 2018. *Content Analysis: An Introduction to Its Methodology*. SAGE Publications.

Mikhail Kuklin and Nikolay Arefyev. 2025. Deepchange at CoMeDi: the cross-entropy loss is not all you need. In *Proceedings of the 1st Workshop on Context and Meaning–Navigating Disagreements in NLP Annotations*, Abu Dhabi, UAE.

Sinan Kurtyigit, Maike Park, Dominik Schlechtweg, Jonas Kuhn, and Sabine Schulte im Walde. 2021. Lexical Semantic Change Discovery. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

Andrey Kutuzov and Lidia Pivovarova. 2021. Rushifteval: a shared task on semantic shift detection for russian. *Komp'yuternaya Lingvistika i Intellektual'nye Tekhnologii: Dialog conference*.

Andrey Kutuzov, Samia Touileb, Petter Mæhlum, Tita Enstad, and Alexandra Wittemann. 2022. NorDiaChange: Diachronic semantic change dataset for Norwegian. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2563–2572, Marseille, France. European Language Resources Association.

Xianming Li and Jing Li. 2023. Angle-optimized text embeddings. *Preprint*, arXiv:2309.12871.

Qianchu Liu, Edoardo Maria Ponti, Diana McCarthy, Ivan Vulić, and Anna Korhonen. 2021. AM2iCo: Evaluating word meaning in context across low-resource languages with adversarial examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7151–7162, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ying Xuan Loke, Dominik Schlechtweg, and Wei Zhao. 2025. ABDN-NLP at CoMeDi shared task: Predicting the aggregated human judgment via weighted few-shot prompting. In *Proceedings of Context and Meaning: Navigating Disagreements in NLP Annotation*, pages 122–128, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Federico Martelli, Najla Kalach, Gabriele Tola, and Roberto Navigli. 2021. SemEval-2021 task 2: Multilingual and cross-lingual word-in-context disambiguation (MCL-WiC). In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 24–36, Online. Association for Computational Linguistics.

J. A. Nelder and R. Mead. 1965. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313.

Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. XL-WSD: an extra-large and cross-lingual evaluation framework for word sense disambiguation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13648–13656. AAAI Press.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Alessandro Raganato, Tommaso Pasini, Jose Camacho-Collados, and Mohammad Taher Pilehvar. 2020. XL-WiC: A multilingual benchmark for evaluating semantic contextualization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7193–7206, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Sentencetransformers (python module). https://www.sbert.net/. Accessed: 2024-10-01.

Julia Rodina and Andrey Kutuzov. 2020. RuSemShift: a dataset of historical lexical semantic change in Russian. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1037–1047, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tetsuya Sakai. 2021. Evaluating evaluation measures for ordinal classification and ordinal quantification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2759–2769.

Gerard Salton and Michael J McGill. 1983. *Introduction to Modern Information Retrieval*. McGraw - Hill Book Company, New York.

Dominik Schlechtweg. 2023. *Human and Computational Measurement of Lexical Semantic Change*. Stuttgart, Germany.

Dominik Schlechtweg, Pierluigi Cassotti, Bill Noble, David Alfter, Sabine Schulte im Walde, and Nina Tahmasebi. 2024. More DWUGs: Extending and evaluating Word Usage Graph datasets in multiple languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14379–14393, Miami, Florida, USA. Association for Computational Linguistics.

Dominik Schlechtweg, Tejaswi Choppa, Wei Zhao, and Michael Roth. 2025. CoMeDi shared task: Median judgment classification & mean disagreement ranking with ordinal Word-in-Context judgments. In *Proceedings of Context and Meaning: Navigating Disagreements in NLP Annotation*, pages 33–47, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Dominik Schlechtweg, Sabine Schulte im Walde, and Stefanie Eckmann. 2018. Diachronic Usage Relatedness (DURel): A framework for the annotation of lexical semantic change. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 169–174, New Orleans, Louisiana.

Dominik Schlechtweg, Nina Tahmasebi, Simon Hengchen, Haim Dubossarsky, and Barbara McGillivray. 2021. DWUG: A large Resource of Diachronic Word Usage Graphs in Four Languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7079–7091, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Karin Schuler. 2005. Verbnet: A broad-coverage, comprehensive verb lexicon.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Víctor Manuel Vargas, Pedro Antonio Gutiérrez, and César Hervás-Martínez. 2020. Cumulative link models for deep ordinal classification. *Neurocomputing*, 401:48–58.

Sachin Yadav. 2025. XL-DURel: Finetuning sentence transformers for ordinal Word-in-Context classification. Master thesis, University of Stuttgart.

Frank D. Zamora-Reina, Felipe Bravo-Marquez, and Dominik Schlechtweg. 2022. LSCDiscovery: A shared task on semantic change discovery and detection in Spanish. In *Proceedings of the 3rd International Workshop on Computational Approaches to Historical Language Change*, Dublin, Ireland. Association for Computational Linguistics.

4: Identical	Identity
3: Closely Related	Context Variance
2: Distantly Related	Polysemy
1: Unrelated	Homonymy

Table 2: The DURel relatedness scale (Schlechtweg et al., 2018) on the left and its interpretation from Schlechtweg (2023, p. 33) on the right.

A Annotation Scale

Unlike WiC, which is designed as a binary classification task, OGWiC employs an **ordinal classification** approach by assigning labels on a **four-point scale**. This four-point scale in Table 2, follows the **DURel annotation framework** proposed by Schlechtweg et al. (2018) which is based on Blank's concept of semantic proximity (Blank, 1997). Unlike GWiC, labels are not transformed post-hoc and each level of the DURel scale has an exact linguistic interpretation as presented in Table 2, where polysemy is located between identity, context variance, and homonymy (Schlechtweg, 2023).

According to Schlechtweg (2023, p. 22–23), the pair (1,2) below is classified as identical as the referents of two uses of the word **arm** are both prototypical representatives of the same extensional category corresponding to the concept 'a human body part':

(1) [...] and taking a knife from her pocket, she opened a vein in her little **arm**, [...]

(2) [...] and though he saw her within reach of his **arm**, [...]

The use pair (1,3) is classified as context variance as both referents still belong to the same extensional category, but one is a non-prototypical representative. Hence, there is some variation in meaning, e.g. the arm of a statue loses the function of the physical arm to be lifted:

(3) [...] when the disembodied **arm** of the Statue of Liberty jets spectacularly out of the sandy beach. [...]

The use pair (1,4) would be classified as polysemy as the two referents of arm belong to different extensional categories, but the corresponding concepts still hold a semantic relation (in this case a similarity relation regarding physical form).

(4) It stood behind a high brick wall, its back windows overlooking an **arm** of the sea [...]

Dataset	LG	Reference	JUD	VER	KRI	SPR
ChiWUG	ZH	Chen et al. (2023)	61k	1.0.0	.60	.69
DWUG	EN	Schlechtweg et al. (2021)	69K	3.0.0	.63	.55
DWUG Res.	EN	Schlechtweg et al. (2024)	7K	1.0.0	.56	.59
DWUG	DE	Schlechtweg et al. (2021)	63K	3.0.0	.67	.61
DWUG Res.	DE	Schlechtweg et al. (2024)	10K	1.0.0	.59	.7
DiscoWUG	DE	Kurtyigit et al. (2021)	28K	2.0.0	.59	.57
RefWUG	DE	Schlechtweg (2023)	4k	1.1.0	.67	.7
DURel	DE	Schlechtweg et al. (2018)	6k	3.0.0	.54	.59
SURel	DE	Hätty et al. (2019)	5k	3.0.0	.83	.84
NorDiaChange	NO	Kutuzov et al. (2022)	19k	1.0.0	.71	.74
RuSemShift	RU	Rodina and Kutuzov (2020)	8k	1.0.0	.52	.53
RuShiftEval	RU	Kutuzov and Pivovarova (2021)	30k	1.0.0	.56	.55
RuDSI	RU	Aksenova et al. (2022)	6k	1.0.0	.41	.56
DWUG	ES	Zamora-Reina et al. (2022)	62k	4.0.1	.53	.57
DWUG	SV	Schlechtweg et al. (2021)	55K	3.0.0	.67	.62
DWUG Res.	SV	Schlechtweg et al. (2024)	16K	1.0.0	.56	.65

Table 3: Datasets used for the CoMeDi shared task. All are annotated on the DURel scale. Spearman and Krippendorff values for RuShiftEval are calculated as average across all time bins. 'LG' = Language; 'JUD' = Number of judgments; 'VER' = Dataset version; 'KRI' = Krippendorff's α; 'SPR' = Weighted mean of pairwise Spearman correlations; 'Res.' = Resampled.

Language	Train	Dev	Test
German	8,279	1,663	3,141
English	5,910	863	2,444
Swedish	5,457	871	1,245
Chinese	10,833	2,532	3,240
Spanish	4,821	621	1,497
Russian	8,029	1,126	2,285
Norwegian	4,504	611	1380
Total	**47,833**	**8,287**	**15,332**

Table 4: Number of data instances per language and split for the OGWiC subtask after cleaning.

In contrast, the referents of arm in the homonymic pair (1,5) belong to different extensional categories and the corresponding concepts do not hold a semantic relation:

(5) And those who remained at home had been heavily taxed to pay for the **arms**, ammunition; fortifications, [...]

B CoMeDi Data

Table 3 shows the source datasets used for the CoMeDi shared task. Table 4 shows the number of data instances per language and split for the OGWiC subtask after cleaning.

C Thresholds

Find the thresholds for mapping cosine similarity to ordinal labels for AnglE-WiC+CoMeDi and XL-LEXEME in Table 5. These were induced on the dev data as described in Section 4.4.

D Spearman Results

Figure 3 shows model performances measured with Spearman correlation and no thresholding.

Dataset	Language	AnglE-WiC+CoMeDi			XL-LEXEME		
CoMeDi	ZH	.577	.677	.793	.495	.650	.655
CoMeDi	EN	.325	.483	.612	.418	.607	.682
CoMeDi	DE	.330	.465	.600	.339	.565	.651
CoMeDi	NO	.210	.339	.488	.390	.414	.522
CoMeDi	RU	.255	.491	.615	.249	.511	.749
CoMeDi	ES	.297	.521	.628	.212	.455	.788
CoMeDi	SV	.290	.452	.564	.419	.646	.672
MCL-WiC	AR		.634			.741	
MCL-WiC	ZH		.766			.814	
MCL-WiC	EN		.668			.705	
MCL-WiC	FR		.623			.667	
MCL-WiC	RU		.594			.775	
WiC	EN		.551			.752	

Table 5: Thresholds for AnglE-WiC+CoMeDi and XL-LEXEME.

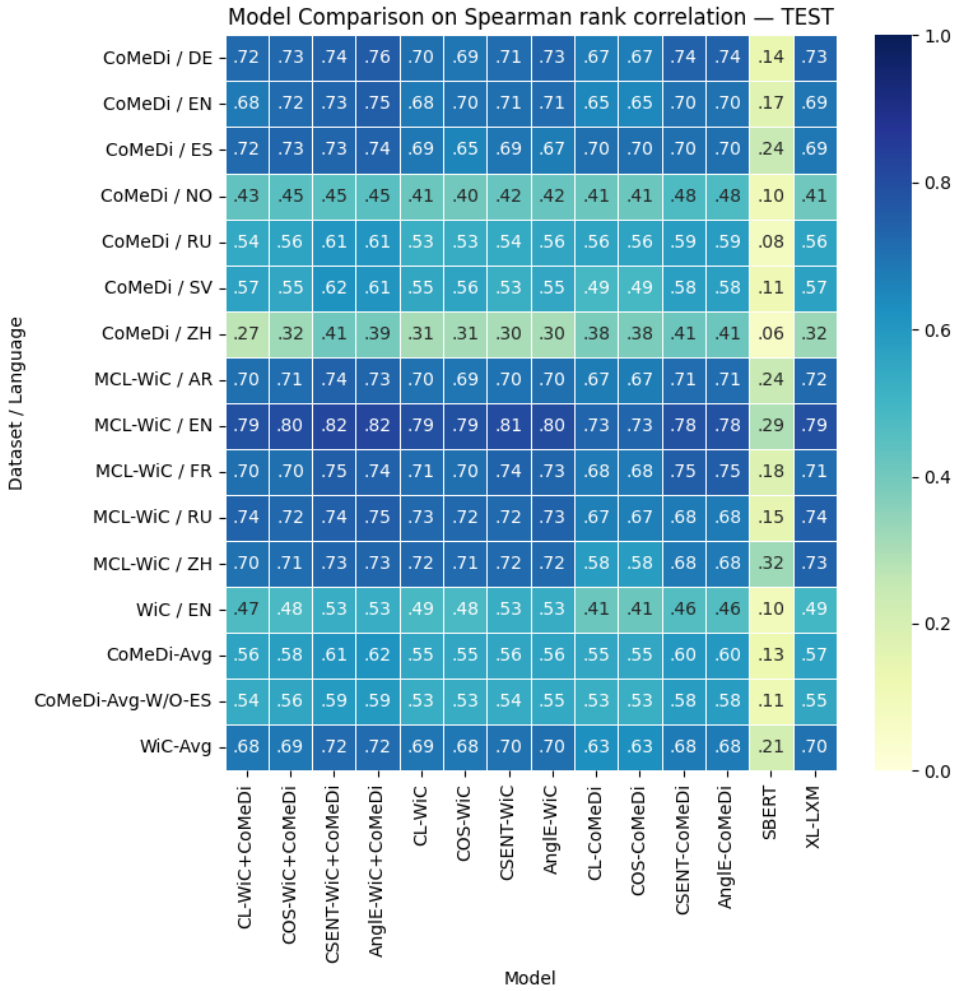

Figure 3: Model evaluation with Spearman's ρ on binary and ordinal test data. CL = Contrastive Loss, COS = Cosine Similarity Loss, CSENT = CoSENT Loss, AnglE = AnglE Loss, XL-LXM = XL-LEXEME.