Predicting Median, Disagreement and Noise Label in Ordinal Word-in-Context Data

Tejaswi Choppa[1] **Michael Roth**[2] **Dominik Schlechtweg**[1]
[1]University of Stuttgart [2]University of Technology Nuremberg
st180670@stud.uni-stuttgart.de michael.roth@utn.de
schlecdk@ims.uni-stuttgart.de

Abstract

The quality of annotated data is crucial for Machine Learning models, particularly in word sense annotation in context (Word-in-Context, WiC). WiC datasets often show significant annotator disagreement, and information is lost when creating gold labels through majority or median aggregation. Recent work has addressed this by incorporating disagreement data through new label aggregation methods. Modeling disagreement is important since real-world scenarios often lack clean data and require predictions on inherently difficult samples. Disagreement prediction can help detect complex cases or to reflect inherent data ambiguity. We aim to model different aspects of ordinal Word-in-Context annotations necessary to build a more human-like model: (i) the aggregated label, which has traditionally been the modeling aim, (ii) the disagreement between annotators, and (iii) the aggregated noise label which annotators can choose to exclude data points from annotation. We find that disagreement and noise are impacted by various properties of data like ambiguity, which in turn points to data uncertainty.

1 Introduction

Machine Learning (ML) research frequently gathers data from human annotators for training and testing of models. It is highly desirable to have good quality data (Sun et al., 2017), because with a low quality of data, the model tends to also learn biases and errors, thereby depreciating model performance. In the process of annotation, usually every instance in the dataset is annotated by multiple annotators in order to reduce the bias of any individual annotator (Uma et al., 2021b). These multiple annotations are subsequently adjudicated to establish a single **gold** label using several descriptive statistical methods. However, using these methods means also discarding the disagreements between annotators, resulting in a loss of information. Recent works propose to include these disagreements into the label aggregation process, treating disagreements as part of the **signal** rather than **noise** (Plank et al., 2014). We take these ideas to the extreme by focusing only on the disagreements and completely ignoring the labels in the aggregation process. The final aim being to construct ML models able to predict the human disagreement on an annotated text instance. Practically, our model may be used to predict instances with high disagreement allowing further modeling components to abstain from predicting the label in order to reduce the error rate (Xin et al., 2021).

For our experiments, we choose the task of semantic proximity annotation involving to quantify how much the meanings of two uses "have in common" (Schlechtweg, 2023). Each of the usage pairs is judged by multiple annotators based on a graded annotation schema. Word senses do not have clear boundaries and often do not fall into disjoint categories (Hanks, 2000; Kilgarriff, 1977) leading to inherent ambiguity. Another often overlooked aspect of data is the data noise. While it is a related phenomenon to disagreement, data noise represents cases where annotators cannot confidently assign labels or instances don't fit predefined categories. Some guidelines address this by offering special exclusion labels (Schlechtweg et al., 2023; Hätty et al., 2019).

Disagreement and noise have a common source: ambiguity. That is, although disagreement and noise are not completely determined by ambiguity, we hypothesize that ambiguity strongly influences these two variables (Uma et al., 2021b; Schlechtweg, 2023). Additionally, we construct more traditional models to predict the aggregated label enabling a comparison with noise and disagreement predictions. Finally, all three modeling approaches can be combined together into one model predicting different important aspects: the aggregated label, the expected disagreement, and

Proceedings of Context and Meaning: Navigating Disagreements in NLP Annotation, pages 65–77
January 19, 2025.

the noisiness of the data point.

2 Related work

In this section, we offer an overview of the previous research on semantic proximity, disagreements and noise in annotation tasks and discuss the methods to include this disagreement into the label aggregation process.

2.1 Tasks on Disagreement detection

NLP tasks often handle disagreements by discarding them or using label aggregation methods. Dawid and Skene (1979) proposed a probablistic label aggregation method calculating posterior probability of labels based on annotator reliability. Sheng et al. (2008) extended this by introducing an uncertainty-preserving labeling scheme that retains disagreement information as probability distributions. Uma et al. creates soft labels from annotator distributions through methods such as standard normalization, the softmax function, and probabilistic label aggregation techniques like MACE, enabling the model to learn from the distribution of annotations.

Although these approaches capture the distribution of disagreeing annotations, there is no significant research on directly predicting the amount of disagreement in a supervised way.

2.2 Research on disagreement for word meaning annotation tasks

Natural Language Processing (NLP) text-based meaning annotation tasks involve assigning semantic (meaning-related) labels to text sequences. Often, this sequence is restricted to a particular word in a context (word usage).

2.2.1 Word Sense Disambiguation

Word Sense Disambiguation (WSD) asks to assign sense glosses to word usages. Glosses are usually taken from a lexical resource like a dictionary or WordNet (Navigli, 2009, p. 2). Erk et al. (2013, p. 3) compare the traditional annotation schema with the possibility of employing graded sense assignments for Word Sense Disambiguation (WSD). The traditional WSD assigns the single most applicable sense from a predefined inventory. On the contrary, the proposed graded schema asks to rate the applicability of each sense on a scale. They discuss theories stating that word senses have "fuzzy boundaries", leading to inherent ambiguity and annotator disagreements. Erk et al.[p. 6] state that people have differences in how concepts and word meanings are mentally represented, causing annotators to assign word senses differently. Erk et al. present graded scales for meaning annotation, moving from the traditional binary annotation scheme. They propose WSsim, where annotators rate the applicability of each WordNet sense for the target lemma on a scale of 1 (sense does not apply) to 5 (sense applies fully).

2.2.2 Semantic proximity

Semantic proximity asks to measure how much meanings of word uses have in **common** (cf. Schlechtweg, 2023, p. 22). Various human annotation studies incorporate semantic proximity by formulating the task as usage similarity (Erk et al., 2013, p. 9) or the semantic relatedness (Schlechtweg et al., 2023, p. 33). Semantic proximity is usually annotated on scales such as the DURel (Schlechtweg et al., 2018) and the USim (Erk et al., 2013, p.9) scales. For the USim task, annotators compare pairs of usages on a five-point similarity scale where 1 means the usages are completely different in meaning and 5 means they are identical in meaning, additionally they permitted the response "cannot decide". For the study of diachronic usage relatedness (DURel), Schlechtweg et al. adopt a relatedness scale similar to that of Brown (2008). For this task, the annotators are asked to choose semantic relatedness between word usage pairs. Refer to Table 1 for the semantic relatedness scale. The label 0 is used when the annotators are unable to make a decision as to the degree of relatedness in meaning between the two word usages e.g. if the sentence is too flawed to understand it, or the meaning of the target word is ambiguous.

4: Identical
3: Closely Related
2: Distantly Related
1: Unrelated

Table 1: The DURel relatedness scale (Schlechtweg et al., 2018).

Tasks and Models of Semantic Proximity The Word in Context (WiC) task introduced by Pilehvar and Camacho-Collados asks to predict the label as TRUE or FALSE based on the similarity of the word usage meanings. On the contrary, the graded WiC task introduced by Armendariz et al. asks to predict the change in the similarity ratings

of a pair of words when the human annotators are presented with an identical pair of words in two distinct contexts and assign a similarity rating for each pair of usages. Leveraging the above two tasks, Zhang (2023) introduces an Ordinal Graded WiC task (OGWiC), which asks to provide labels on an ordinal scale from 1 to 4 following the relatedness scale from the DURel framework (Schlechtweg et al., 2018). For the WiC, GWiC and the OGWiC tasks, the main methodology employed by models is similar and it involves feeding an input string to the contextual embedder, creating one or more vector representations. Then, the vector processor post-processes the embeddings e.g. by concatenation or using cosine similarity. The resulting embedding is then passed to a classification head for WiC or a ranker for GWiC or through an ordinal classifier for OGWiC (Zhang, 2023). Pilehvar and Camacho-Collados use the contextualized word embedding models like Context2Vec, ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) to compute dynamic word representations based on the context, on top of which classifiers like a simple Multi-layer perceptron (MLP) and a threshold-based classifier using vector cosine similarity are used. As GWiC has a multi-lingual dataset, most submissions utilise Cross-Lingual Model XLM-R (Conneau et al., 2019), a multi-lingual version of RoBERTa for the embedder part. Additionally, Zhang (2023) employs DistilBert and XL-Lexeme (Cassotti et al., 2023) embedders and these embeddings are processed as vectors by concatenating the embeddings, getting the cosine similarity of word embeddings and Hadamard product of word embeddings. Zhang (2023) employs a nominal classification head that treats the ordinal regression task as a standard multi-class classification problem.

3 Tasks

Given a pair of word usages, we aim to predict three data properties: (i) median semantic proximity, (ii) the level of disagreement, and (iii) the presence of noise in Word-in-Context annotations. We will treat each of these aspects in a separate task. The first two tasks have been included into the recently organized CoMeDi task (Schlechtweg et al., 2025). Each instance consists of a target word w, two usage contexts $c1$ and $c2$ expressing specific meanings of w, and multiple semantic proximity ratings by annotators on a scale of 1 (completely unrelated meanings) to 4 (identical meanings), following the DURel annotation framework. As an example, consider the word usages below (Schlechtweg, 2023, pp. 22–23), from which we build annotation instances by combining them into usage pairs.

(1) ... and taking a knife from her pocket, she opened a vein in her little **arm**, and dipping a feather in the blood, wrote something on a piece of white cloth, which was spread before her...

(2) ... and though he saw her within reach of his **arm**, yet the light of her eyes seemed as far off...

(3) It stood behind a high brick wall, its back windows overlooking an **arm** of the sea which, at low tide, was a black and stinking mud-flat...

(4) ...the company decided to create a new **arm**

The use pair (1,2) with sample judgments [4, 4] would likely receive semantic proximity label 4.0 (identical) as both refer to a physical human arm. The use pair (1,3) with sample judgments [2,3,2] would be classified as **polysemy** as the two referents of *arm* belong to different extensional categories (human arm vs. arm of the sea), but the corresponding concepts still hold a semantic relation (in this case a similarity relation regarding physical form). This pair would rather receive a lower label such as 2.0 (distantly related). In contrast, the *arm* in the **homonymic** pair (1,4) with sample judgments [1, 0, 0], belong to different extensional categories and it's relatively harder to determine if the corresponding concepts hold a semantic relation, especially in the context 4 (could mean weapon or branch of company). This pair would receive a noise label of 1.0 with semantic proximity and disagreement labels being NaN.

Ordinal Graded Word-in-Context (OGWiC) requires predicting the median of annotator judgments for each use pair, formulated as an ordinal classification task and evaluated using Krippendorff's α (Krippendorff, 2018). Treating graded WiC as an ordinal classification task instead of a ranking task constrains model predictions to exactly reproduce instance labels instead of just inferring their relative order (Schlechtweg et al., 2025). This is advantageous if ordinal labels have an interpretation because predictions then inherit this

interpretation. Such an interpretation can be assigned to the DURel scale as explained above, like for the example pair (1,2) with sample judgments [4,4], the median semantic proximity label 4 can be interpreted as *identity* which means the meanings of the word in both the contexts are identical.

Disagreements in Word-in-Context (DisWiC) asks to predict the mean of pairwise absolute judgment differences between annotators:

$$D(J) = \frac{1}{|J|} \sum_{(j_1,j_2)\in J} |j_1 - j_2|,$$

where J is the set of unique pairwise combinations of judgments. For pair 1, 3 it amounts to

$$D(J) = \frac{1}{3}\left(|(2-3)| + |(2-2)| + |(3-2)|\right) = 0.67$$

DisWiC can be formulated as a ranking task based on the magnitude of disagreement and evaluated using Spearman's ρ (Spearman, 1904).

Noise Word in Context (NoiseWiC) asks to predict the noise in the data annotations. It is formulated as a binary classification task, which is evaluated with the nominal version of Krippendorff's α and Accuracy. The noise label is calculated as follows:

$$N(J) = \begin{cases} 1, & \text{if (\# non-zero } < \text{\# zero)} \\ \text{NaN}, & \text{if (\# non-zero} \geq \text{\# zero)} \\ & \text{\& (\# zero} > 0) \\ 0, & \text{otherwise} \end{cases}$$

For pair (1,4) from above, N(J) = 1 since there are more '0' annotations than the non-zero annotations.

3.1 Evaluation

OGWiC involves the classification task of detecting the semantic proximity label. Since the labels are of ordinal nature, we will use Krippendorff's α (Krippendorff, 2018), which, in its ordinal formulation, 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 regression (Sakib et al., 2023). For the DisWiC task, since the output has continuous disagreement labels, we will use Spearman's correlation (Spearman, 1904) as our evaluation measure because it helps capture non-linear relationships better. NoiseWiC is a binary classification task, so we mainly rely on accuracy as the classification metric but we also report the Krippendorff's α for nominal data (Krippendorff, 2018).

4 Data

For all our tasks, we make use of publicly available ordinal WiC datasets from the CoMeDi shared task (Schlechtweg et al., 2025), as summarized in Table 7 in Appendix A. These provide a large number of judgments for use pairs from different datasets across different languages annotated on the DURel scale and have so far not been used primarily for WiC-like tasks, but only for semantic change detection purposes.

4.1 Data aggregation and cleaning

For cleaning and aggregation, the Shared Task organizers initially exclude annotation instances with less than two annotations (Schlechtweg et al., 2025). For OGWiC, then instances with any 0-judgments ("Cannot decide") and instances with any pair of annotators disagreeing more than one point on the annotation scale are discarded. The organizers then calculate the median of all judgments, for each instance. Instances with a non-integer median (e.g. 3.5) are discarded. For all remaining instances, gold labels are given by the median of judgments. For DisWiC, the organizers derive instance labels by aggregating over judgments with the average of pairwise absolute annotator deviations, as discussed in section 3. 0-judgments ignored in this process. For NoiseWiC, we assign a noise label of 1 (indicating the presence of noise) if the number of 0-judgments by annotators exceeds the number of non-zero judgments. Otherwise, a label of 0 is assigned to indicate that noise is not present.

For each of the tasks, the organizers then randomly split the target words per language into train/test/dev with sizes of 70/20/10%. In contrast to previous tasks, the organizers intentionally do not balance out the label distribution in order to keep more realistic data conditions. Find an overview of the final splits per language in Table 2.

5 Models

For all our tasks, we follow a similar model architecture, except for the classification head. For this, we aim to utilize the best-performing models from WiC, GWiC, and ordinal GWiC, as discussed in Section 2.2.2, particularly with embedders, since

Task	# Instances	# Uses	Split
OGWiC	48K	55K	Train
	8K	8K	Dev
	15K	16K	Test
DisWiC	82K	55K	Train
	13K	8K	Dev
	26K	16K	Test
NoiseWiC	204K	55K	Train
	32K	8K	Dev
	64K	16K	Test

Table 2: Data statistics after cleaning and aggregation per split and and over all languages combined.

all tasks share a common focus on pairwise in-context meaning annotation. We use them as follows:

Contextual Embedder Given the input word usages, we employ XL-Lexeme, as it was optimized on binary WiC datasets and is one of the top-performing models for the OGWiC task, as noted by Zhang. XL-Lexeme (Cassotti et al., 2023) is a bi-encoder model utilizing a Siamese Network that extends the Sentence-BERT (SBERT) architecture to focus on the target word within input sentences. The model is trained using a contrastive loss function, which minimizes the cosine distance between the encoded representations when the target word has the same meaning and maximizes the distance when the meanings differ. It is pretrained on WiC datasets like MCL-WiC (Martelli et al., 2021), AM2ICO (Liu et al., 2021), and XL-WiC (Raganato et al., 2020), enabling it to function similarly to sentence-level encoders, while specifically focusing on target words marked using special tokens (<t> and </t>) to emphasize their context. This approach allows the model to better identify whether the target word maintains the same meaning across different contexts. Given an input sentence and the position of the target word (start and end character indices of the word within the sentence), XL-Lexeme generates a contextualized embedding for the target lemma in context.

Vector Processor We use the word embeddings as input to different models in different ways. For the CosTH model (see below), we use the embedding vectors of the words in two contexts and take their cosine similarity, based on which the thresholds are optimized. For all other models, we concatenate the embeddings of both words in context to create a single representation. This approach is useful when employing a classifier that takes the full feature set into account, such as a Multi-layer Perceptron (MLP). (Pilehvar and Camacho-Collados, 2019).

Classification Head Based on the nature of the task, we use different classification or regression approaches. For the OGWiC task, we use the following classification heads:

- `Cosine + threshold (CosTH)`: Given two vector representations of different contexts, we use a threshold-based classifier that utilizes the cosine similarity between the vectors. For these cosine similarity values, the classifier optimizes thresholds per language by minimizing a custom loss function, Krippendorff's α in our case, to determine the labels as follows:

$$\text{minimize} \quad L(\mathbf{y}, \hat{\mathbf{y}}|\theta) = 1 - \alpha(\mathbf{y}, \hat{\mathbf{y}}_\theta),$$

  where $\mathbf{y}, \hat{\mathbf{y}}$ are gold labels and predicted cosine similarities respectively, α is Krippendorff's α and $\hat{\mathbf{y}}_\theta$ is a mapping of cosine similarities to thr ordinal labels based onthresholds θ. We optimize thresholds per language.

- `Linear Regression (LR)`: The Linear Regression (Pedregosa et al., 2012) predicts continuous distribution of values by optimizing the Mean Square Error between a linear combination of the features and the ground truth. In our case, given the concatenated vector as input, Linear Regression predicts continuous values. The semantic proximity labels on a scale of 1 to 4 are then mapped from these predicted continuous values based on pre-defined thresholds based on rounding to the next integer, see Equation 5.

$$\text{threshold}(y_{\text{pred}}) = \begin{cases} 1, & \text{if } y_{\text{pred}} < 1.5 \\ 2, & \text{if } 1.5 \leq y_{\text{pred}} < 2.5 \\ 3, & \text{if } 2.5 \leq y_{\text{pred}} < 3.5 \\ 4, & \text{else} \end{cases} \tag{5}$$

- `Multilayer Perceptron (MLP)`: A Multilayer Perceptron (MLP) (Rosenblatt, 1958) is a feedforward artificial neural network that

learns complex patterns and perform tasks like classification and regression. It consists of input layers, hidden layers and a output layer. In the WiC task (Pilehvar and Camacho-Collados, 2019), this approach has been used as a baseline. Given the concatenated vector as input, we use the MLP classifier to predict the semantic proximity label. We try to optimize the batch size, activation function, hidden layer size and alpha parameters, see Table 8 in appendix A.

For the DisWiC task, we use the following classification heads:

- `Multilayer Perceptron (MLP)`: Given the concatenated vectors as input, we use the MLP regressor which unlike the MLP Classifier model use MSE loss function and linear activation function to predict the the continuous disagreement labels. We optimize the batch size, activation function, hidden layer size and alpha parameters as well along with early stopping to prevent the model from overfitting.

- `Linear Regression`: Given the concatenated vector as input features, we use Linear Regression to predict the continuous disagreement values.

For the NoiseWiC task, we use logistic regression as a classification head. Logistic regression is a model used for binary classification tasks (Cox, 2018), predicting the probability that a given input belongs to either of the classes. It uses a sigmoid function to map predictions to a 0-1 probability range. Logistic regression generally uses 0.5 as a threshold value to map the probabilities to the binary labels. The probabilities greater than or equal to 0.5 are mapped to the label 1 and vise versa. In our case, given the concatenated vector as input, the logistic regression model is used detect the "noise" label.

5.1 Upperbound Metric

We explore an upperbound metric, which refers to the maximum performance a model can achieve on a given task. The main aim of an Upperbound is to set the model performance into context and to understand what we can expect from the model's performance and provide context to that performance. Model performance is typically expected to fall between the baseline and the Upperbound. For the OGWiC task, we compute the Upperbound metric by iteratively calculating Krippendorff's α between a single, excluded annotation and the median label built from the remaining annotations, weighted by their share of total annotations. For the DisWiC task, we compute the Upperbound metric by iteratively calculating Spearman's rank correlation between the mean disagreement of an excluded annotator pair and the mean disagreement label derived from the remaining annotations. Instances must have at least four annotators, as the definition of the disagreement measure requires at least two annotators for its calculation.

5.2 Baseline Models

5.2.1 Baseline XLM-R embedder

`XLM-R` : XLM-R (eXtreme Language Model Roberta) is an extension of RoBERTa that uses self-supervised training techniques to achieve state-of-the-art performance in cross-lingual understanding. It has been used as the underlying language model for fine-tuning XL-Lexeme (Cassotti et al., 2023). It was trained to learn robust representations from large-scale multilingual data (Conneau et al., 2019).We use the boolean mask to identify and extract subwords corresponding to the target token, extract the corresponding embeddings for the target subwords, and aggregate them using mean pooling to obtain the target token embedding. It is paired with CosTH model and Linear Regression model as classification heads for the OGWiC and DisWiC tasks respectively.

5.2.2 Majority Baseline

For the NoiseWic task, we employ a majority class baseline that assigns the most frequently occurring class in the train dataset to every instance in the test dataset, which, in our case, was the '0' label. This baseline provides a minimum performance threshold that a model should exceed.

5.2.3 Feature Baseline

The model architecture employs embedding features from pre-trained language models, as is common in many semantic NLP tasks (Pilehvar and Camacho-Collados, 2019; Schlechtweg et al., 2020). We engineer a set of simple linguistic features that correlate with noise or disagreement, including lexical complexity, grammatical complexity, and context richness.For the DisWiC task, we engineer features such as character length and the presence of non-alphabetic characters in the con-

text to evaluate their impact on performance, using an MLP to predict disagreement labels.

6 Experiments

Our experiments aim to predict a median semantic proximity, mean disagreement or noise label based on the input usages. We experiment with different components of our model and compare their performances for this task. Also, we explore the factors influencing the disagreements through our experiments. The code for these experiments is available online.[1]

For generating the contextualized word embeddings, we primarily use XL-Lexeme, with XLM-R serving as the baseline model. For each of the subtasks, the models are fit on the training data in two ways: (i) per language, i.e., hyperparameters or thresholds are learned per language, and (ii) on the entire training data available. We experiment with various hyperparameter values for different classification heads across different tasks, see Appendix A. For OGWiC task, we used the default parameters for the scikit-learn linear regression model, as there were very few tunable parameters `n_jobs`, `fit_intercept`, and `copy_X`. Similarly, for the Cosine+Threshold model (CosTH), we implement it without hyperparameter tuning due to the lack of tunable parameters. In case of the DisWiC task, for the Linear Regression classification head, we again use the default parameters. For the MLP model in both OGWiC and DisWic tasks, we perform grid search over the specified hyperparameter grid, see Table 8 in Appendix A, fitting the model with each combination on the training data and evaluating its performance on the development data using the Spearman correlation as scoring metric. It keeps track of the best-performing combination and outputs the best score and hyperparameters at the end. We choose the hyperparameters for our grid by relying on Pilehvar and Camacho-Collados (2019), who use a solver 'Adam', batch size of 32 and hidden layer size 100. We take these values and expand our grid. We also take default parameters of the scikit-learn MLP in the parameter grid.

Apart from that, we also included some parameters like the hidden layer size and learning rate from Chai et al. (2021, p. 6). We also give standard scaler as an option in the parameter grid to improve the overall performance of the MLP. The

[1] https://github.com/choppa98/Supervised-semantic-proximity-detection

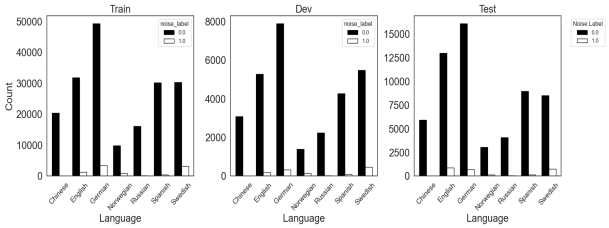

Figure 1: Label distribution for NoiseWiC task per language.

NoiseWiC dataset, refer Figure 1 is highly skewed with the majority label being 0. Especially, in languages like chinese and russian there is little to no presence of the noise label 1.0, as you can see in the Figure 4. In order to address this class imbalance and to avoid any bias associated with it, we omit instances of these languages while carrying out our experiments. Also, we employ a sampling strategy to downsample the majority class to match the size of the minority class. After downsampling, each language group is a balanced dataset with equal number of instances from both classes, see Figure 5 in Appendix A.

Apart from this, we also conduct an error analysis on the disagreed instances. We went through annotator comments for various patterns of disagreements and manually inspect the corresponding contexts to understand reasons for '0' annotations and annotator disagreements.

7 Results and Analysis

For the OGWiC task, as shown in Table 3, the Cosine+Threshold model achieves the best performance among classification heads, with an average Krippendorff's α of 0.67 on the development data and 0.58 on the test data. While XL-Lexeme + MLP shows relatively high performance in the "All data' setting (α of 0.55 on development, 0.42 on test), it performs lower in the "Per Lang" setting (α of 0.37 on development, 0.28 on test). The baseline model, XLM-R with Cosine+Threshold, underperforms (α of 0.25 on development, 0.12 on test). XL-Lexeme+Linear Regression performs poorest across all languages and settings. For ZH (Chinese), models generally performed better on development data but poorly on test data. NO (Norwegian) shows consistently low performance across models. For EN (English), both XL-Lexeme+CosTH and XL-Lexeme+MLP achieve moderately high results, while for DE (German) and SV (Swedish), XL-Lexeme+CosTH performs

particularly well.

For the DisWiC task, the XL-Lexeme+MLP model shows best performance with average Spearman's ρ of 0.16 on development and 0.15 on test data in the "All Data" setting. In the "Per Lang" setting, it achieves 0.16 on test data but only 0.11 on development data. The MLP model shows high variability across languages, with ZH and NO achieving higher Spearman's ρ on test data. ES (Spanish) and EN exhibited consistently low values across settings and splits. The Linear Regression model yielded lower results (Spearman's ρ of 0.11 on development, 0.10 on test in "All data" setting). The baseline XLM-R+Linear regression model gives similar average Spearman's ρ as XL-Lexeme under "All data" setting, with ZH showing relatively higher ρ on both development and test data. The Upperbound metric for DisWiC provides inaccurate comparison results due to insufficient data for Chinese and Norwegian i.e, each instance in these languages has been annotatated by less than four annotators. For the NoiseWiC task, XLM-R+Logistic Regression achieves best results with average accuracy of 0.62 on development and 0.59 on test data (Krippendorff's α of 0.24 and 0.14 respectively). XL-Lexeme+Logistic Regression achieves 0.58 on both sets, performing particularly well for EN and SV. ES consistently shows lower scores, similar to the DisWiC task.

Model	Setting	Split	AVG	ZH	EN	DE	NO	RU	ES	SV
XLM-R + CosTH	Lang	Dev	.25	.51	.17	.3	.03	.27	.44	.05
		Test	.12	.06	.10	.27	.12	.11	.17	.02
XL-Lexeme + CosTH	Lang	Dev	**.67**	**.77**	**.66**	**.75**	**.52**	.62	.62	**.75**
		Test	.58	.38	.65	.72	.51	.55	**.65**	.60
XL-Lexeme + LR	All	Dev	.20	.37	.09	.33	.20	.05	.24	.15
		Test	.16	.04	.26	.15	.06	.15	.26	.18
	Lang	Dev	.10	.11	.19	.31	-.08	.13	-.13	.19
		Test	.09	.06	.04	.15	.03	.22	.22	-.07
XL-Lexeme + MLP	All	Dev	.55	.63	.49	.65	.48	.47	.48	.68
		Test	.42	.35	.49	.39	.37	.44	.51	.40
	Lang	Dev	.37	.17	.17	.60	.24	.32	.50	.59
		Test	.28	.20	.36	.36	.23	.32	.34	.13
Upperbound	All	Dev	.96	1.	.97	.92	.97	.95	.96	.96
		Test	.95	1.	.97	.88	.94	.96	.96	.95

Table 3: Krippendorff's α for OGWiC task. All = 'All Data', Lang = 'Per Lang'.

8 Analysis

As we observe in Section 7, for the OGWiC task, the models show promising results with highest α being 0.67 on development data and 0.58 on test data. But the models performed rather poorly on the DisWiC and the NoiseWic tasks. In case of the DisWiC task, the number of annotators significantly impacted performance, with Chinese and Norwegian having few annotators (most instances annotated by less than four). For analyzing annotator disagreement levels (0.66, 1.33, 3.0), instances from the English dataset reveal that fewer annotators can lead to more consistent labeling as their variation becomes more predictable.

For the instances with the disagreement label 3.0, among the annotators, it was observed that most of the disagreements, see example in Appendix A, occurred in the presence of a "0" label which corresponds to the "cannot decide" label. Another common pattern observed was that the highly disagreed instances had mostly two annotators whose judgments were [1, 4], which means either annotator agrees that the meaning of the target lemma is identical in both the contexts or completely unrelated. This pattern originates from the task definition: Only in the case of two annotators the maximum disagreement score of 3.0 can be reached. Generally, more annotators lead to a decrease in the score. This is because, with more annotators, the pairwise distances between some individual labels must be either small or zero, resulting in lesser maximal possible disagreement.

For example, refer to Appendix A, that had a mean disagreement of 1.33, the annotator judgments varied with all the annotators mostly having unique judgment per instance. In the cases, see Example 10 in Appendix A, where a mean disagreement of 0.66 was observed, the judgments mostly corresponding to a pattern of only one annotator disagreeing with the rest of the group. Key factors affecting these disagreement levels include grammatical errors, misspelled words, lack of context, and complex language misinterpretation. Likewise, annotator uncertainty in many cases raises questions about annotator reliability in meaning annotation tasks. Additionally, on analyzing various noise patterns, the background knowledge about various domains also determined the annotator's assignment of the '0' label. All these factors indicate the influence of the underlying data properties, such as ambiguity, which in turn point to data uncertainty.

Model	Setting	Split	AVG	ZH	EN	DE	NO	RU	ES	SV
XLM-R + LR	All	Dev	.11	.31	.07	.16	.12	.05	.02	.07
		Test	.11	.38	.06	.09	.07	.04	.07	.08
	Lang	Dev	.02	.01	-.05	.09	.07	-.01	-.01	.04
		Test	.05	.10	.01	.13	.04	.11	.05	-.11
Feature Baseline	All	Dev	-.00	.03	-.04	.01	-.05	.02	.00	.02
		Test	-.00	-.00	-.00	.00	-.03	-.01	-.01	.02
XL-Lexeme + LR	All	Dev	.11	.16	.01	.06	.26	.002	.03	.21
		Test	.10	.30	.02	.03	.06	.07	.05	.18
	Lang	Dev	.10	.11	.19	**.31**	-.08	.13	-.13	.19
		Test	.09	.06	.04	.15	.03	**.22**	**.22**	-.07
XL-Lexeme + MLP	All	Dev	.16	.36	.03	.11	.33	.06	.05	.15
		Test	.15	.45	**.07**	.07	.10	.13	.08	.16
	Lang	Dev	.11	.06	.04	.11	**.35**	.04	-.02	**.23**
		Test	**.16**	**.48**	.04	.11	.25	.04	.06	.16
Upperbound	All	Dev	.16		-.09	.16		.32	.21	.20
		Test	.18		.07	.04		.22	.08	.48

Table 4: Spearman's ρ for DisWiC task.All = 'All Data', Lang = 'Per Lang'.

Model	Split	AVG	EN	DE	NO	ES	SV
Majority Baseline	Dev	.5	.5	.5	.5	.5	.5
	Test	.5	.5	.5	.5	.5	.5
XLM-R +Logistic Reg	Dev	**.62**	.57	**.67**	**.70**	.55	.65
	Test	.59	.59	.65	.47	**.60**	.63
XL-Lexeme +Logistic Reg	Dev	.58	**.61**	.59	.55	.48	**.68**
	Test	.58	.59	.63	.58	.48	.63

Table 5: Accuracy for NoiseWiC task.

9 Conclusion

In this study, we have formulated the OGWiC task, the DisWiC and the NoiseWiC task. We focus on predicting semantic proximity, disagreement, and noise labels using contextualized word embeddings across multiple languages. For OGWiC, the combination of XL-Lexeme with a Cosine + Threshold approach achieved the highest Krippendorff's α scores of 0.67 on the development data and 0.58 on the test data. In DisWiC, the MLP classification head significantly outperformed Linear Regression, particularly when trained per language, with hyperparameter tuning enhancing performance in languages like Chinese and Norwegian. NoiseWiC had challenges due to class imbalance, especially in languages with sparse noise labels, which we addressed through downsampling; however, model performance remained low, as indicated by the Krippendorff's α scores. Across tasks, XL-Lexeme consistently outperformed the baseline XLM-R, especially in language-specific setups. Training strategies: whether using all data or per language, played a crucial role, with per-

Model	Split	AVG	EN	DE	NO	ES	SV
XLM-R +Logistic Reg	Dev	**.24**	.09	**.33**	**.39**	.10	.29
	Test	.14	.17	.30	-.21	**.20**	.25
XL-Lexeme +Logistic Reg	Dev	.13	**.21**	.16	.06	-.11	**.36**
	Test	.15	.19	.27	.15	-.08	.26

Table 6: Krippendorff's α for NoiseWiC task.

language tuning improving performance. Further, our analysis of results lays a stepstone for future work especially for the DisWiC task.

Limitations

When instances are annotated by different numbers of people, it becomes tricky to make direct comparisons of disagreement levels between those instances. Take two cases: when an instance has two annotators versus three annotators, the maximum possible disagreement between them will be inherently different. This variation in annotator numbers may help explain why we see different performance patterns across languages. For instance, the Chinese dataset stands out because every instance in this language has been annotated by two annotators. Going forward, we should do two things: first, explore alternative ways to measure disagreement, and second, ensure that all instances receive the same number of annotations to make comparisons more meaningful. Also, for the noise detection, the high imbalance in labels especially for Russian and Chinese pose a challenge.

Acknowledgments

This work is strongly based on a master thesis written at the Institute for Natural Language Processing at the University of Stuttgart (Choppa, 2024). Work by Michael Roth was funded by the DFG Emmy Noether program (RO 4848/2-1). Dominik Schlechtweg has been funded by the research program 'Change is Key!' supported by Riksbankens Jubileumsfond (under reference number M21-0021).

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.

Carlos Santos 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 Twelfth Language Resources and Evaluation Conference*, pages 5878–5886, Marseille, France. European Language Resources Association.

Susan Windisch Brown. 2008. Choosing sense distinctions for WSD: Psycholinguistic evidence. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 249–252, Stroudsburg, PA, USA.

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.

Soo See Chai, Whye Lian Cheah, Kok Luong Goh, Yee Hui Robin Chang, Kwan Yong Sim, and Kim On Chin. 2021. A multilayer perceptron neural network model to classify hypertension in adolescents using anthropometric measurements: A cross-sectional study in sarawak, malaysia. *Computational and Mathematical Methods in Medicine*, 2021(1):2794888.

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. 2024. Supervised semantic proximity noise and disagreement detection. Master thesis, University of Stuttgart.

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*.

D. R. Cox. 2018. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232.

A. P. Dawid and A. M. Skene. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):20–28.

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. Association for Computational Linguistics.

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

Patrick Hanks. 2000. Do word meanings exist? *Computers and the Humanities*, 34(1/2):205–215.

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.

Adam Kilgarriff. 1977. What is word sense disambiguation good for? In *Proc. Natural Language Processing in the Pacific Rim (NLPRS '97)*, Phuket, Thailand.

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

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.

Qianchu Liu, Edoardo M Ponti, Diana McCarthy, Ivan Vulić, and Anna Korhonen. 2021. Am2ico: Evaluating word meaning in context across low-resource languages with adversarial examples. *arXiv preprint arXiv:2104.08639*.

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.

Roberto Navigli. 2009. Word sense disambiguation: a survey. *ACM Computing Surveys*, 41(2):1–69.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2012. Scikit-learn: Machine learning in python. *CoRR*, abs/1201.0490.

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*, pages 2227–2237, New Orleans, LA, USA.

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.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Linguistically debatable or just plain wrong? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume volume 2, pages 507–511. 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. *arXiv preprint arXiv:2010.06478*.

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.

Frank Rosenblatt. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408.

Fardin Ahsan Sakib, A H M Rezaul Karim, Saadat Hasan Khan, and Md Mushfiqur Rahman. 2023. Intent detection and slot filling for home assistants: Dataset and analysis for Bangla and Sylheti. In *Proceedings of the First Workshop on Bangla Language Processing (BLP-2023)*, pages 48–55, Singapore. Association for Computational Linguistics.

Dominik Schlechtweg. 2023. *Human and Computational Measurement of Lexical Semantic Change*. Ph.D. thesis, University of Stuttgart, 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*, Miami, Florida. Association for Computational Linguistics.

Dominik Schlechtweg, Tejaswi Choppa, Wei Zhao, and Michael Roth. 2025. The CoMeDi shared task: Median judgment classification & mean disagreement ranking with ordinal word-in-context judgments. In *Proceedings of the 1st Workshop on Context and Meaning–Navigating Disagreements in NLP Annotations*, Abu Dhabi, UAE.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for 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.

Dominik Schlechtweg, Shafqat Mumtaz Virk, Pauline Sander, Emma Sköldberg, Lukas Theuer Linke, Tuo Zhang, Nina Tahmasebi, Jonas Kuhn, and Sabine Schulte im Walde. 2023. The durel annotation tool: Human and computational measurement of semantic proximity, sense clusters and semantic change. *Preprint*, arXiv:2311.12664.

Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, page 614–622, New York, NY, USA. Association for Computing Machinery.

Charles Spearman. 1904. The proof and measurement of association between two things. *American Journal of Psychology*, 15:88–103.

Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. 2017. Revisiting unreasonable effectiveness of data in deep learning era. *Preprint*, arXiv:1707.02968.

Alexandra Uma, Tommaso Fornaciari, Anca Dumitrache, Tristan Miller, Jon Chamberlain, Barbara Plank, Edwin Simpson, and Massimo Poesio. 2021a. SemEval-2021 task 12: Learning with disagreements. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 338–347, Online. Association for Computational Linguistics.

Alexandra Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2021b. Learning from disagreement: A survey. *Journal of Artificial Intelligence Research*, 72:1385–1470.

Ji Xin, Raphael Tang, Yaoliang Yu, and Jimmy Lin. 2021. The art of abstention: Selective prediction and error regularization for natural language processing. 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 1040–1051, Online. Association for Computational Linguistics.

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.

Tuo Zhang. 2023. An ordinal formulation of the graded word-in-context task. Master thesis, University of Stuttgart.

A Appendix

(6) `Context 1`: The public, gene- /z/ **rally**, remained indifferent, notwithstanding the marvellous things which were related of the terri tory which had been ceded to the company.

(7) `Context 2`: Once or twice I have known him touch nerves that go close to the heart; but gene **rally**, he is no master of the feelings.

`Observation`: misspelled, grammatically incorrect

`Judgments` : [1, 0, 0, 4]

`Mean Disagreement Label` : 3.0

`Comments available` : ”, ’same word, but incomplete’, ’generally?’, ’UNK; I think the intended meaning of the target word might be generally in both sentences’]

(8) `Context 1`: Willoughby’s as the family possess and will submit for examination, carefully searched, in the hope that some

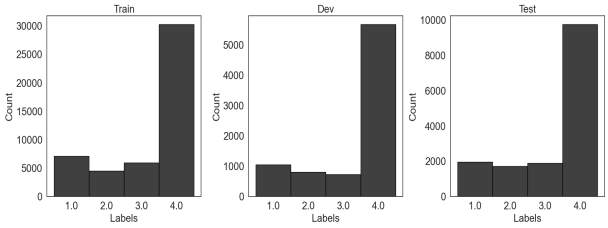

Figure 2: Label Distribution for OGWiC task.

record may be found in his hand-writing, sufficiently clear to establish the fact that my mother was the wife of the elder Captain Allen.

(9) `Context 2`: For the **record**, your information is inaccurate on Governor Rockefeller’s visit on Sept. 21.

`Judgments` : [3, 4, 2]

`Mean Disagreement Label` : 1.333

`Comments available` : [”, ’If “for the record” is used metaphorically and not literally in sentence 2, then a rating of 3 would be more appropriate.’, ”] On other front,

`Observation`: Context 1 talks about a physical record like a book or document whereas Context 2 refers to stating a fact or information.

(10) `Context 1`: Ari arrived at Kibbutz Revivim Tuesday **afternoon**, at the peak of the sun’s arc across

`Context 2`: Old shopping lists and ticket stubs and wads of listed newsprint come falling around Pafko in the faded **afternoon**.

`Judgments` : [3, 4, 4]

`Mean Disagreement Label` : 0.66

`Observation` : Both refer to the mid day time frame, also referring to how the afternoon looks like

`Comments available` : [’daylight versus actual day’, ”, ”]

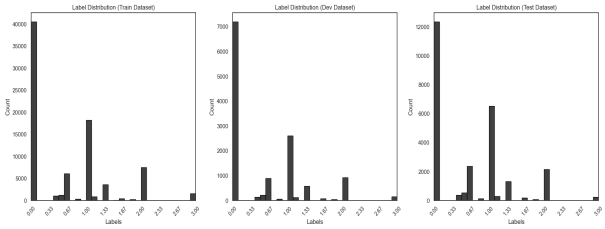

Figure 3: Label distribution for DisWiC task.

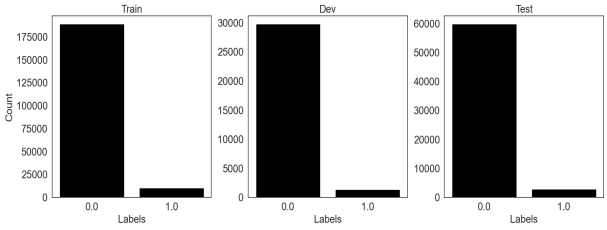

Figure 4: Label distribution for NoiseWiC task.

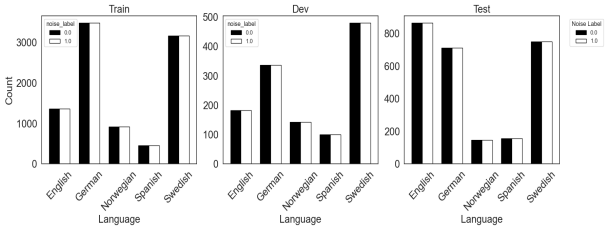

Figure 5: Label distribution for NoiseWiC task per language after downsampling.

Dataset	LG	Reference	JUD	VER	KRI	SPR
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
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	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
ChiWUG	CH	Chen et al. (2023)	61k	1.0.0	.60	.69
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
NorDiaChange	NO	Kutuzov et al. (2022)	19k	1.0.0	.71	.74

Table 7: Datasets used for our task. All are annotated under 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.

Parameter	Values
activation	relu, tanh
solver	Adam
hidden layer sizes	10, 50, 100
alpha	0.0001, 0.001, 0.01, 0.1
batch size	32, auto, 50, 100
scaler	StandardScaler(), None

Table 8: Parameter grid used for tuning MLP.

Hyperparameter	Model	Task	ZH	EN	DE	NO	RU	ES	SV
Activation	MLP	OGWiC	relu	relu	relu	relu	relu	relu	relu
Alpha			.0001	.0001	.0001	.0001	.0001	.0001	.0001
Batch Size			10	10	10	10	10	10	10
Hidden Layers			(10,)	(10,)	(10,)	(10,)	(10,)	(10,)	(10,)
Scaler			yes	yes	yes	None	yes	yes	yes
Solver			adam	adam	adam	adam	adam	adam	adam
Activation	MLP	OGWiC	tanh	tanh	relu	relu	tanh	tanh	tanh
Alpha			.001	.1	.0001	.001	.1	.1	.0001
Batch Size			auto	auto	auto	100	100	auto	100
Hidden Layers			(50,)	(50,)	(50,)	(50,)	(100,)	(100,)	(100,)
Scaler			None	yes	yes	yes	yes	yes	yes
Solver			adam	adam	adam	adam	adam	adam	adam

Table 9: Final set of hyperparameters for MLP per task in 'Per Lang' setting.

Hyperparameter	Model	Task	Value
Activation	MLP	OGWiC	relu
Alpha			.0001
Batch Size			auto
Hidden Layers			(100,)
Scaler			None
Solver			adam
Activation	MLP	DisWiC	relu
Alpha			.001
Batch Size			auto
Hidden Layers			(50,)
Scaler			yes
Solver			adam

Table 10: Final set of hyperparameters for MLP in 'All Data' setting