Lexical Adversarial Negative Examples for Word Sense Disambiguation

Anonymous ACL submission

Abstract

Fine-grained word meaning resolution remains a major challenge for Language Models (LMs), which often rely on global sentence representations and consequently fail to capture subtle local semantic distinctions. To address this limitation, we propose LANE, a novel adversarial training strategy that shifts the model's learning focus toward the target word. LANE generates challenging negative training examples through an adversarial procedure that dynamically queries the model to select the alternative marked word whose representation is most similar to the anchor. This process encourages the model to increase separability between identical sentences containing different marked words, thereby promoting more discriminative word representations. Experiments on word sense disambiguation benchmarks show that LANE consistently improves performance over standard contrastive learning baselines. Qualitative analyses further indicate that the proposed adversarial negatives help the model capture subtle meaning differences, even in challenging contexts. The method is model-agnostic and can be readily integrated into existing representation learning frameworks.

1 Introduction

Word Sense Disambiguation (WSD), the task of identifying the precise meaning of a word in context, remains central to deep semantic understanding. It underpins applications such as Lexical Semantic Change Detection (LSCD) (de Sá et al., 2024) and word similarity tasks (Armendariz et al., 2020). For example, WSD systems aim to determine whether the word "crazy" denotes insanity (as in *That's a crazy man*) or excitement (as in *That's crazy, man*), and whether "bank" refers to a financial institution or a river edge. Although such words appear in similar contexts, their usage can signal entirely different meanings.

Modern LMs like XLM-Roberta achieve impressive results in this task, but their final word representations often overfit to global context rather than encoding a word's specific sense (Liu et al., 2021; Vulić et al., 2023; Ethayarajh, 2019). They solve the task by capturing topical cues, for instance, inferring that "loan" and "interest" signal a financial sense of "bank", but can misrepresent a word's fine-grained contribution (Xu et al., 2025; McCoy et al., 2019). Such failures to capture precise contextual meaning hinder performance in downstream tasks requiring genuine semantic nuance, like neologism identification (McCrae, 2019).

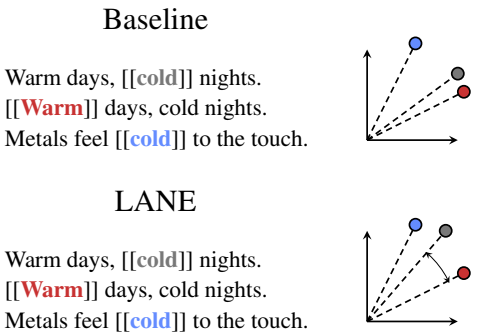

Figure 1: Embedding behavior under standard contrastive training (top) versus LANE (bottom). Colors mark the anchor (gray), a same-sense positive in a different context (blue), and an adversarial negative obtained by re-marking a different word in the anchor sentence (red).

A widely adopted strategy to improve lexical sensitivity in WSD is target-word highlighting, where the word of interest is marked (e.g., surrounding with special tokens) before being encoded by a language model (Cassotti et al., 2023). The underlying assumption is that explicit marking encourages the model to attend more directly to the lexical semantics of the word. Yet, as our analysis reveals (Figure 1), this assumption is fragile: embeddings of the same sentence remain nearly identical regardless of which word is highlighted, indicating

that the model often resolves the task using only the surrounding context (Liu et al., 2021). For instance, the sentences "<t>big</t> or small?" and "big or <t>small</t>?", marking the antonyms 'big' and 'small', produce highly similar vectors[1].

In practice, this results in nearly indistinguishable representations for semantically distinct words, while instances of the same sense across different contexts may be mapped to more distant vectors. Such behavior illustrates the persistence of shortcut learning (Robinson et al., 2021), where models exploit superficial contextual regularities rather than grounding their predictions in the lexical meaning of the marked token.

This imbalance produces a collapsed representation space, where embeddings concentrate in a narrow region irrespective of the highlighted word, what we call **lexical insensitivity** (Section A). The consequence is reduced lexical separability and weak generalization on tasks requiring sensitivity to word-level sense, especially in contexts that are semantically similar. Taking this into consideration, in this paper, we investigate *How can we ensure that the representation space meaningfully reflects the contribution of the target word, instead of sentence information?*

To address this, we introduce the Lexical Adversarial Negative Example (LANE) framework, which biases representations toward lexical relevance. LANE generates adversarial negatives by dynamically substituting the highlighted word with a candidate that yields the most similar representation to the anchor, while keeping the sentence context unchanged. This forces the model to resolve genuine adversarial challenges rather than contextual variation: the hard negatives share the full contextual frame of the anchor but differ in the target token, and thus in lexical semantics.

By presenting nearly identical sentences that differ only in the marked word, LANE sharpens semantic boundaries and encourages the model to attend to the lexical contribution of the target word, yielding embeddings that better capture fine-grained meaning and exhibit desirable properties of lexical semantics (Section A).

2 Related Work

Representing fine-grained word meaning has long been a challenge in natural language processing. Early approaches tackled this problem through lexical resources such as WordNet (Miller, 1995) and the creation of sense-annotated corpora to support supervised WSD methods (Raganato et al., 2020; Bevilacqua et al., 2021; Huang et al., 2024a). With the advent of deep learning, contextualized word embeddings such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) became standard, enabling substantial gains in WSD and related tasks by dynamically adjusting representations according to context.

Yet despite these advances, contextualized models often fail to distinguish subtle word-level meaning differences, tending instead to overfit to sentence-level semantics (Ethayarajh, 2019). This weakness is particularly visible in LSCD, which requires tracking the semantic shifts of individual words across time and domains. While static embedding approaches offered initial baselines (Schlechtweg et al., 2019; Martinc et al., 2020), more recent work has leveraged contextualized embeddings to capture dynamic variation (Schlechtweg et al., 2020; Kutuzov and Giulianelli, 2020; Giulianelli et al., 2020). Nevertheless, isolating the semantics of the target word from the broader discourse remains an open challenge.

To address these limitations, recent research has turned to contrastive learning as a way to induce more discriminative semantic representations. Inspired by advances in computer vision (Chen et al., 2020), contrastive objectives have been adapted to NLP, yielding stronger sentence and word embeddings for tasks such as semantic similarity and clustering (Gao et al., 2021; Chuang et al., 2022).

Within lexical semantics, several directions have emerged. One line of work leverages multilingual pretraining on large-scale, diverse corpora to improve contextual coverage of word senses (Cassotti et al., 2023; Yadav and Schlechtweg, 2025). Another incorporates auxiliary supervision, such as in-context sense induction, to encourage sense separation (Giulianelli et al., 2023; Mosolova et al., 2024; Liétard et al., 2024; Li'etard and Loiseau, 2025). A third explores data augmentation, for example by altering input structure: Martelli et al. (2021) proposed swapping sentence order in a cross-encoder, though this strategy does not extend to bi-encoder architectures where contrastive losses are typically computed.

Complementary to these efforts, adversarial strategies aim to construct more challenging negatives to prevent representational collapse. Prior

[1] Cosine similarity $= 0.84$ using XL-LEXEME (State-of-the-Art), senses > 0.75 are considered identical by this model.

work has explored lightweight perturbations such as case alternation (Wang et al., 2022), synonym and antonym substitution (Wang et al., 2021), and token replacements with masked language model predictions (Chuang et al., 2022). Multilingual adversarial signals have further been obtained through cross-lingual links in Wikipedia, which align English sentences with low-resource counterparts while introducing cross-lingual semantic contrasts (Liu et al., 2021). While these methods increase robustness and sharpen decision boundaries, they typically operate at the sentence level, focusing on global semantic differences rather than directly enforcing separability of meanings for a specific word in context.

Our work addresses this gap by combining the strengths of contrastive learning and adversarial augmentation while targeting their shared limitation: the absence of adversarial signals that operate at the word level. Instead of relying on heuristic perturbations or external lexical resources, we generate adversarial negatives by marking different words within the same sentence, producing pairs that are contextually identical yet lexically distinct. This strategy is model-agnostic and integrates seamlessly into existing representation learning frameworks, complementing multilingual pretraining and auxiliary-task sense induction while providing adversarial challenges directly tied to the phenomenon of interest: the fine-grained disambiguation of word meaning.

3 Datasets

Evaluating language models on a single dataset has long been standard practice in Natural Language Processing (NLP), yet these evaluations often provide an incomplete view of generalization and robustness (Lones, 2024). Even within a single task, datasets may differ substantially in domain, linguistic complexity, annotation conventions, and underlying assumptions. Consequently, relying on one benchmark risks overfitting to dataset-specific idiosyncrasies rather than assessing a model's true capacity for semantic discrimination in WSD.

To obtain a more comprehensive evaluation, we curated three datasets covering diverse sources and temporal spans: SEMCOR (Miller et al., 1993), MASC[2], and FEWS (Blevins et al., 2021). These datasets differ primarily along two dimensions: (i) time of creation (i.e. 1993 vs. 2020) and (ii) source type (i.e. dictionaries, books, blogs). This diversity enables controlled variation in language register and data quality (ranging from the standardized style of dictionaries to the more informal, conversational tone of blogs) thus exposing models to a broad spectrum of lexical and contextual phenomena.

Additional to the created datasets we evaluate our models in MCL-WiC data (Martelli et al., 2021) and DWUG (Schlechtweg et al., 2021). While MCL-WiC and DWUG are formatted for WSD formulations, SEMCOR, MASC, and FEWS required preprocessing as they just present usages and sense keys. We generated contrastive pairs by merging instances with identical lemma and POS tags, then pairing them across distinct contexts. This process yielded pairs of the same lexical item in different environments, supporting fine-grained semantic discrimination. Ground-truth labels were automatically assigned using sense keys: pairs sharing the same key were labeled as positive (1), while those differing in sense were labeled as negative (0). This alignment to sense inventories ensures that the resulting pairs reflect genuine semantic distinctions rather than superficial contextual variation.

For SEMCOR, MASC, and FEWS, we partitioned data lexicographically such that test sets contain words beginning with letters "P" or later. This reduces lexical overlap between training and testing splits, forcing models to generalize to unseen lexical items. Unlike DWUG and WiC, our resources include adjectives and adverbs in addition to nouns and verbs, thereby broadening the coverage of syntactic categories and semantic phenomena.

For multilingual evaluation, we adopt XL-WiC (Raganato et al., 2020) which extends WiC's word in context formulation to multiple languages (see Table 5). XL-WiC draws on both WordNet and Wiktionary to maintain cross-lingual consistency. Additionally, we train with XL-LEXEME, a composite resource integrating XL-WiC, MCL-WiC, and AM2iCO, to further enhance cross-lingual and cross-categorical representativeness. Table 2 illustrates positive, negative, and the hard-negative pairs LANE generates by varying the marked token within identical sentence contexts.

4 Methodology

Let V be a token vocabulary and $s = (w_1, \ldots, w_k) \in V^*$ a sentence in which a target

[2] https://anc.org/data/masc/

Table 1: Examples of sentence pairs in the proposed datasets.

Dataset	Label	Sentence 1	Sentence 2
SEMCOR (1993)	T	The heavens opened, pelting them with **hail** the size of walnuts.	Drought, **hail**, disease, and insects take their toll of crops.
MASC (2003)	F	The report wasn't hard, I had already read the book, and so I just jotted down a few **notes** for him.	Passing **notes** when the teacher isn't paying attention.
FEWS (2021)	T	Virchow stated that premature fusion of this suture results in **pachycephalic** deformity.	[...] that a judge may be learned in the law, but woefully **pachycephalic** in matters scientific.

Table 2: Examples of positive, negative, and adversarial pairs in WiC. Adversarial pairs are generated by LANE.

Label	Sentence 1	Sentence 2
positive	Sound <t>**carries**</t> well over water.	Many metals <t>**carry**</t> heat.
negative	Sound <t>**carries**</t> well over water.	You must <t>**carry**</t> your camping gear.
lane	Sound <t>**carries**</t> well over water.	Sound carries well over <t>**water**</t>.

position i is delimited by markers $\langle t\rangle\, w_i\, \langle /t\rangle$. An encoder

$$f_\theta : V^* \times \{1, \ldots, k\} \to \mathbb{R}^d$$

produces a target-token representation, here the mean-pooled contextual hidden states over the marked span. The intent is that $f_\theta(s, i)$ encode the sense of w_i as used in s: the representation should be context-conditioned (so polysemous w in different sentences may map differently) and lexically grounded (so distinct targets in the same sentence map differently).

The pathology we observe is the latter failing. For typical s and $i \neq j$,

$$f_\theta(s, i) \;\approx\; f_\theta(s, j),$$

so the representation factors through the sentence alone, $f_\theta(s, i) \approx \tilde{g}_\theta(s)$. We call this *lexical insensitivity*: the encoder reduces to a sentence summarizer that ignores which token has been designated as the target. The mechanism is straightforward: each contextual hidden state in a Transformer attends globally, so mean-pooling over any subspan, including a single-token span, yields a vector dominated by sentential context rather than local lexical identity. For WiC this is fatal: given w appearing in s_1 and s_2 at positions i_1, i_2, the decision rule depends on $\langle f_\theta(s_1, i_1), f_\theta(s_2, i_2)\rangle$; under marker invariance this collapses to $\langle \tilde{g}_\theta(s_1), \tilde{g}_\theta(s_2)\rangle$, measuring sentence similarity rather than sense agreement.

Standard contrastive objectives, including the ball-packing view of Robinson et al. (2020), do not preclude this regime. Positive pairs (same sense, different sentences) and negative pairs (different sense, different sentences) both vary the surrounding context, so an encoder $\tilde{g}_\theta$ that ignores the marker can satisfy the contrastive constraints whenever same-sense sentences are themselves more similar than different-sense ones, a property the training distribution often supplies for free.

We close this loophole with LANE that hold the context fixed and vary only the target. Given an anchor (s, i), construct

$$s' \;=\; (w_1, \ldots, \langle t\rangle, w_j, \langle /t\rangle, \ldots, w_k), \quad j \neq i$$

where j is chosen by maximizing target-representation similarity over a candidate pool $j \in \mathcal{C} = \{1, \ldots, k\}$ drawn from the training corpus:

$$j \;=\; \arg\max_{j \in \mathcal{C}} \; \mathrm{sim}\big(f_\theta(s, i),\, f_\theta(s', j)\big).$$

See Algorithm 1. Because s and s' differ only at marking position i, a lexically insensitive encoder yields $\tilde{g}_\theta(s) \approx \tilde{g}_\theta(s')$, since the encoder is smooth over the small variance introduced by the additive positional embeddings of the <t> tags translation, and hence $f_\theta(s, i) \approx f_\theta(s', j)$, incurring maximal loss on the adversarial pair. The construction is in effect a probe: the higher the attained sentence similarity, the more lexical-invariant the encoder. Mixing these adversarial negatives into the contrastive objective therefore forces f_θ to break lexical invariance and attend to w_i specifically, concentrating gradient on the very distinction WiC is built to test.

4.1 Adversarial Negative Examples

Bi-encoders independently represent vectors in the embedding space without explicit awareness of which target word the model should attend to (Liu et al., 2021). Although marking tokens are intended to signal the target, they have minimal effect in practice (Section 5). To improve word-sense learning, we aim to reduce the model's reliance on contextual cues during training. To this end, we generate adversarial negative examples with two desirable properties:

- **Property 1:** A contextualized word representation should produce a distinct vector whenever the marked lexical item changes, even if the surrounding context remains identical.
- **Property 2:** For words X in sentence A and Y in sentence B that share the same sense, their embeddings should remain similar only when X and Y are the marked items; the embeddings should diverge whenever a different lexical item is marked in A or B.

Building on these properties, we generate adversarial examples that explicitly encode this learning constraint. During training of each dataset, we construct: (i) pairs of identical sentences that differ only in which word is marked as the target; and (ii) pairs in which sentences A and B contain the same sense for a given word, but a different word in A is marked. This construction ensures that even identical or highly similar contexts yield distinct representations when the lexical target changes.

In Algorithm 1, we illustrate our method for dynamically generating negative examples in-batch. The input consists of a dictionary containing two target words (w_1, w_2), two corresponding sentences (s_1, s_2), a similarity label (l), and a similarity threshold (h):

- First, the function extracts all tokenized words from s_1 and filters out any occurrences of w_1.
- For each remaining candidate w, the model computes $f_\theta(w, s_1)$, where the sentence text is left unchanged and only the marker is moved to the position of w. The candidate that maximizes the cosine similarity between this re-marked embedding and the original $f_\theta(w_1, s_1)$ is selected as w'.
- If the resulting similarity exceeds threshold h and w' is not a synonym of w_1, the output replaces w_2 with w' and sets s_2 identical to s_1, so the two sentences differ only in which token is marked.
- Otherwise the example is kept unchanged.

The newly generated example is assigned label zero, serving as a challenging negative: because the contexts are identical, the model must rely on the marked lexical item itself. Synonyms are filtered using a separate annotation step in which Gemma-3 (12B) labels candidate pairs (w', w_1) as synonymous or not, avoiding the multilingual coverage limitations of WordNet.

Input: $w_1, s_1, w_2, s_2, l, h, f_\theta$
$\mathcal{W} \leftarrow \text{tokenize}(s_1)$;
$\mathcal{C} \leftarrow \{\, w \mid w \in \mathcal{W},\ w \neq w_1 \,\}$;
$sim(w) \leftarrow \cos(f_\theta(w, s_1), f_\theta(w_1, s_1))$;
$w' \leftarrow \arg\max_{w \in \mathcal{C}} sim(w)$;
if $h < \cos(f_\theta(w', s_1), f_\theta(w_1, s_1))$ ***and*** *not_synonym*(w', w_1) **then**
  $w_2 \leftarrow w'$;
  $s_2 \leftarrow s_1$;
  $l \leftarrow 0$;
end
return w_1, s_1, w_2, s_2, l;

Algorithm 1: Pseudo-code for creating adversarial negative examples (`lexical-negative`)

We restrict the adversary to tokens present in s_1 to ensure the adversarial negative remains grammatically coherent and semantically grounded in the same discourse context. Substitutions from the full vocabulary risk producing incoherent sentences that introduce confounds unrelated to lexical disambiguation. The procedure thus implements an approximate min-max:

$$\min_{\theta} \max_{w' \in \mathcal{C}} \cos(f_\theta(w', s_1), f_\theta(w_1, s_1)) \qquad (1)$$

The inner maximization selects the token in $\mathcal{C}$ whose marked embedding is closest to that of w_1 (the hardest negative), and the outer minimization trains θ to push this pair apart through the contrastive loss. To prevent the model from overfitting to these challenging negatives or collapsing into a local optimum, we introduce the adversarial examples gradually through a scheduled insertion strategy.

4.2 Scheduler

Introducing adversarial negatives too early in training risks convergence to suboptimal local minima

(Section C), as the signal difference is too subtle to handle at this stage. In such cases, the model may overfit to a narrow subset of challenging comparisons before developing robust and generalizable representations (Xuan et al., 2020); in our experiments, early introduction collapsed the representation space.

To mitigate this issue, we employ a linear scheduling strategy, scaled by the training epoch, to gradually introduce adversarial negatives. During the initial warm-up phase of 5 epochs, the model is trained without any adversarial examples, allowing it to form stable base representations. After this stage, the proportion of adversarial examples per batch increases linearly with each epoch, reaching a final ratio of 0.2. We retain only candidates whose similarity exceeds the threshold $h = 0.8$, ensuring that the kept negatives are genuinely hard. All negatives are computed dynamically from in-batch data rather than precomputed, enabling the model to adapt continuously as training evolves.

Our ablation studies corroborate the effectiveness of this strategy (Section C): the linear scheduler prevents early collapse, enhances training stability, and achieves superior overall performance compared to settings where adversarial examples are introduced prematurely or at a fixed rate.

4.3 Optimization Objective

To differentiate senses, we optimize our models with a contrastive learning objective based on SimCSE (Gao et al., 2021). Following recent literature in lexical differentiation, we employ an in-batch, cosine-based loss (Yadav and Schlechtweg, 2025). This formulation leverages implicit ranking within each batch, thereby promoting semantically coherent clustering via cosine similarity. Our choice aligns with prior state-of-the-art methods and provides a strong, empirically validated baseline. All models are optimized using the CoSENT loss (Huang et al., 2024b), as formalized in the equation below:

$$\mathcal{L} = \log\left(1 + \sum_{i,j \;\; y_i < y_j} e^{\lambda(s_i - s_j)}\right) \qquad (2)$$

Here, s_k denotes the cosine similarity score for the k-th embedding pairs such that the expected similarity of s_i is greater than s_j. The summation extends over all ordered pairs in the batch such that $y_i < y_j$. If the embeddings are u_k and v_k, then $s_k = \cos(u_k, v_k) = \frac{u_k \cdot v_k}{\|u_k\|\|v_k\|}$. y_k represents the ground-truth similarity label for the k-th pair (1=true, 0=false), and λ is a trainable scaling factor that controls the sharpness of the distribution.

If the model ranks correctly ($s_j > s_i$), the difference $(s_i - s_j)$ is negative, and the exponential term $e^{\lambda(s_i - s_j)}$ becomes small, contributing minimally to the overall loss. Conversely, when the model ranks incorrectly ($s_i > s_j$), the difference $(s_i - s_j)$ is positive, causing $e^{\lambda(s_i - s_j)}$ to grow large and yield a higher loss value. This formulation thus penalizes misranked pairs more heavily while suppressing contributions from correctly ranked ones. In all experiments, we use the standard scaling parameter $\lambda = 20$.

5 Evaluation

We assess the effectiveness of the proposed LANE method on the datasets described in Section 3. Each dataset is split into train, development, and test partitions. During training, negative mining is applied exclusively to encourage more discriminative word representation, while the development split is used for model selection, and the test split is reserved for final evaluation. For SEMCOR, MASC, and FEWS we have only test sets, so we use the models trained on MCL-WiC to test generalization.

To quantify the added value of LANE, we conduct controlled comparisons using two strong encoders for semantic similarity models based on SimCSE (Gao et al., 2021) with BERT, MPNet, and the established state-of-the-art XLM-RoBERTa (XLM-R). CLINE and DiffCSE exhibit a strong bias toward sentence-level reconstruction, leading to worse performance than the SimCSE baseline, so we excluded them; how to adapt these models to lexical tasks remains an open question. For each architecture, we evaluate both the baseline model trained with standard contrastive objectives and the same model augmented with LANE. This design isolates the impact of LANE on learning fine-grained word-sense distinctions, independent of the underlying architecture.

All models are trained using AdamW optimizer (Loshchilov and Hutter, 2017) with a learning rate of 1e-5, 500 warm-up steps, a weight decay of 0.01, and an effective batch size of 64. Models are trained for 100 epochs until no improvement on test set (early stopping n=5). This evaluation framework ensures a fair and direct comparison, allowing us to highlight improvements in word-sense repre-

sentation and downstream performance attributable specifically to LANE. We train the models with 5 different seeds and report mean ± standard deviation. In Tables 3–5, bold marks the best result per dataset (column) under a paired t-test at $p < 0.05$; when two configurations are statistically indistinguishable, both are bolded.

Table 3: Comparison of methods across datasets in terms of accuracy on the test split (n=5 seeds). The -SYN ablation disables the synonym filter (LLM-based annotation with Gemma-3 12B).

Model	MCL-WiC	DWUG
Gemma3-12B	66.4 ± 0.7	56.9 ± 0.8
BERT	86.5 ± 0.3	74.4 ± 0.2
BERT+**LANE**-SYN	89.5 ± 1.3	74.7 ± 0.8
BERT+**LANE**	**90.0 ± 0.4**	**75.0 ± 0.2**
MPNet	87.8 ± 0.6	74.1 ± 0.2
MPNet+**LANE**-SYN	90.1 ± 0.4	74.5 ± 0.3
MPNet+**LANE**	**90.6 ± 0.3**	**74.4 ± 0.2**
XLM-R	89.1 ± 0.8	**73.8 ± 0.1**
XLM-R+**LANE**-SYN	90.9 ± 0.8	73.7 ± 0.2
XLM-R+**LANE**	**90.8 ± 0.6**	**73.9 ± 0.5**

As shown in Table 3, incorporating LANE adversarial negatives consistently improves, or at minimum maintains, accuracy across all evaluated datasets and architectures. BERT and MPNet both see notable gains on MCL-WiC, while XLM-RoBERTa, despite its strong multilingual pretraining baseline, improves further with LANE and achieves the highest scores overall. The WordNet-based synonym filter provides a small additional boost over the `-SYN` ablation, but gains are largely retained without it, indicating the method's applicability to languages lacking such resources. For the DWUG data, the gains are diminished compared to MCL-WiC; we attribute this to the restricted coverage of DWUG (only 20 words), as the dataset was not originally designed for WiC.

Table 4: Comparison of methods across OOD datasets in terms of accuracy (n=5).

Model	SEMCOR	MASC	FEWS
BERT	77.1 ± 0.1	77.6 ± 0.2	**89.9 ± 0.3**
BERT+**LANE**	**78.5 ± 0.4**	**81.3 ± 0.6**	**89.5 ± 0.3**
MPNet	77.4 ± 0.4	77.8 ± 0.3	88.8 ± 0.2
MPNet+**LANE**	**79.0 ± 0.3**	**80.9 ± 0.6**	**91.1 ± 0.3**
XLM-R	77.6 ± 0.3	79.2 ± 0.6	91.6 ± 0.6
XLM-R+**LANE**	**78.5 ± 0.3**	**81.1 ± 0.6**	**93.1 ± 0.2**

On the OOD evaluation (Table 4), LANE yields gains on SEMCOR, MASC, and FEWS across all three encoders, suggesting that the lexical signal generalizes beyond the training distribution. While BERT achieves performance comparable to XLM-R on monolingual datasets, its monolingual design restricts applicability in multilingual settings (Table 5).

We further evaluate our approach in a multilingual word sense disambiguation setting using the XL-LEXEME datasets (XL-WiC (Raganato et al., 2020), MCL-WiC (Martelli et al., 2021), AM2ICO (Liu et al., 2021)). We denote our model as XL+LANE (trained in XL-LEXEME data) and have released it on HuggingFace[3]. Table 5 reports our results on the XL-WiC test set, comparing XL-LANE with the base XL-LEXEME model and prior state-of-the-art methods.

As shown in the table, incorporating LANE consistently improves performance across most languages. XL-LANE achieves notable gains in English, French, German, Bulgarian, Chinese, and Dutch, demonstrating that adversarial word-level negatives enhance cross-lingual word-sense discrimination. These results confirm that LANE provides a measurable benefit even when applied to strong multilingual baselines. We further assess our approach in an unsupervised cross-lingual generalization scenario using XL-WiC across nine target languages: Bulgarian (bg), Chinese (zh), Croatian (hr), Danish (da), Dutch (nl), Estonian (et), Farsi (fa), Japanese (ja), and Korean (ko). The model is trained solely on XL-LEXEME data and compared to XL-DUREL, which leverages substantially more training resources. Despite this difference, XL-LANE achieves comparable performance, highlighting its data efficiency and robust cross-lingual generalization. Overall, these experiments extend the findings from the monolingual evaluation: LANE consistently enhances word-level representations, providing performance gains across languages, and supporting effective sense disambiguation in both supervised and cross-lingual settings.

6 Discussion

Our results highlight a persistent limitation of contextual language models: even highly capable encoders often rely on global sentence semantics rather than representing the fine-grained meaning of the target word. Consistent with prior work (Liu et al., 2021), we observe that embeddings of the

[3]Omitted for review.

Table 5: Comparison of classifiers with LANE for supervised multi-lingual (XL-WiC).

Model	en	fr	de	it	bg	zh	hr	da	nl	et	fa	ja	ko	MEAN
XL-LEXEME	72.2	78.5	84.8	**75.6**	82.7	80.2	72.7	76.6	78.2	66.4	66.6	68.2	79.8	75.5
XL-DUREL	73.2	77.8	85.0	72.9	75.4	77.7	**75.2**	75.6	79.2	67.6	**70.6**	**69.7**	80.1	75.3
XL-LANE	**73.4**	**80.3**	**86.8**	74.8	**83.6**	**80.4**	72.5	**77.1**	**79.7**	**68.9**	65.0	68.0	**81.0**	**76.3**

same sentence remain nearly identical regardless of which word is marked (Figure 2), indicating that lexical information contributes little to the learned representation. This reliance on coarse contextual cues undermines interpretability and restricts generalization to new domains or languages where such cues differ.

The proposed LANE framework mitigates this issue by introducing lexically controlled adversarial negatives, sentences that differ only in the marked token. This formulation constrains the learning process to capture distinctions that are attributable to the target word itself. As shown in Figure 2, this results in more structured embedding spaces: sentences containing the same sense are drawn closer together, while those differing in meaning are pushed farther apart. The improved intra-sense cohesion and inter-sense separation explain the consistent accuracy gains reported in the evaluation.

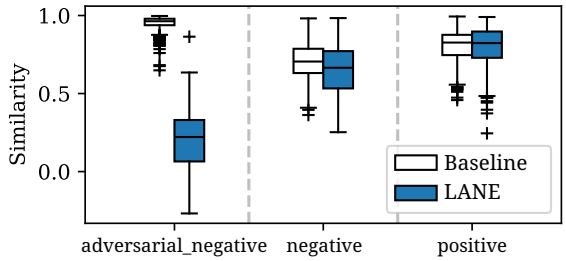

Figure 2: Similarity for representations learned under a traditional regime and under LANE.

Figure 3 plots the average last-layer attention head for the baseline and LANE-trained models, further illustrating that models trained with LANE attend more strongly to the marked token, suggesting that adversarial negatives shift attention away from irrelevant contextual cues and toward the lexical element being disambiguated. This demonstrates that the benefits of LANE are not merely representational but also functional, influencing how the model allocates its focus during inference.

Beyond accuracy gains, our findings provide evidence that tasks purportedly requiring deep semantic understanding have been solved through sentence-level cues (Liu et al., 2021), indicating

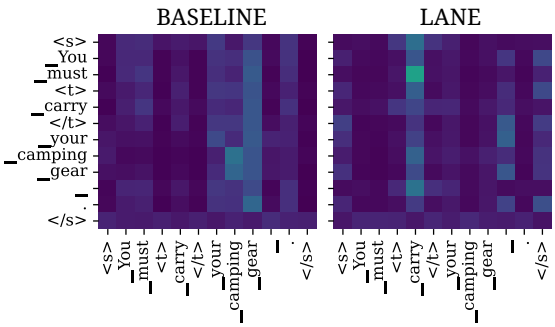

Figure 3: Last attention head heatmap for the baseline and LANE.

that significant open problems persist in both the semantic capabilities and the evaluation of transformer architectures.

7 Conclusion

We presented LANE, a robust adversarial method for generating highly challenging negatives in lexical representation tasks. By focusing on the target word rather than the surrounding context, LANE encourages models to encode fine-grained lexical distinctions, resulting in more robust and discriminative representations. Our experiments demonstrate that LANE consistently improves performance across mono and multilingual WSD benchmarks, including out-of-distribution settings, without requiring complex hyperparameter tuning.

Importantly, we show that traditional approaches using marked words alone often fail to induce representations that prioritize lexical meaning, instead relying heavily on sentence-level contextual cues. LANE addresses this limitation by explicitly enforcing word-level separability, producing embeddings that better reflect true semantic distinctions.

Overall, LANE offers a model-agnostic and effective strategy for enhancing lexical semantic differentiation, improving generalization, and strengthening interpretability. Its straightforward integration into existing frameworks renders it a practical tool for a broad range of lexical representation tasks, including cross-lingual and low-resource scenarios.

Limitations

The LANE method relies on the quality of the tokenizer and the data used for synonym filtering. Reproducing these experiments with a sub-optimal tokenizer or language model for specific languages may lead to performance degradation, for instance, low-resource languages and complex morphology. Although the method should work without explicitly avoiding synonyms (we observed synonyms in fewer than 0.1% of created examples and still saw improvements on WiC without filtering them), we recommend the synonym filter for optimal results.

References

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.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. Recent trends in word sense disambiguation: A survey. In *International Joint Conference on Artificial Intelligence*.

Terra Blevins, Mandar Joshi, and Luke Zettlemoyer. 2021. FEWS: Large-scale, low-shot word sense disambiguation with the dictionary. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 455–465, Online. Association for Computational Linguistics.

Pierluigi Cassotti, Lucia Siciliani, Marco DeGemmis, Giovanni Semeraro, and Pierpaolo Basile. 2023. XL-LEXEME: WiC pretrained model for cross-lingual LEXical sEMantic changE. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1577–1585, Toronto, Canada. Association for Computational Linguistics.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljacic, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022. DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States. Association for Computational Linguistics.

Jader Martins Camboim de Sá, Marcos Da Silveira, and Cédric Pruski. 2024. Survey in characterization of semantic change. *ArXiv*, abs/2402.19088.

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.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. 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 55–65, Hong Kong, China. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3960–3973, Online. Association for Computational Linguistics.

Mario Giulianelli, Iris Luden, Raquel Fernandez, and Andrey Kutuzov. 2023. Interpretable word sense representations via definition generation: The case of semantic change analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3130–3148, Toronto, Canada. Association for Computational Linguistics.

Dandan Huang, Lu Cao, Zhenting Li, and Yue Zhang. 2024a. Which sense dominates multisensory semantic understanding? a brain decoding study. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17557–17563, Torino, Italia. ELRA and ICCL.

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

Andrey Kutuzov and Mario Giulianelli. 2020. UiO-UvA at SemEval-2020 task 1: Contextualised embeddings for lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 126–134, Barcelona (online). International Committee for Computational Linguistics.

Bastien Liétard, Pascal Denis, and Mikaela Keller. 2024. To word senses and beyond: Inducing concepts with contextualized language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2684–2696, Miami, Florida, USA. Association for Computational Linguistics.

Bastien Li'etard and Gabriel Loiseau. 2025. Cale : Concept-aligned embeddings for both within-lemma and inter-lemma sense differentiation.

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.

Michael A. Lones. 2024. Avoiding common machine learning pitfalls. *Patterns*, 5(10):101046.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *ArXiv*, abs/1711.05101.

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.

Matej Martinc, Syrielle Montariol, Elaine Zosa, and Lidia Pivovarova. 2020. Discovery team at SemEval-2020 task 1: Context-sensitive embeddings not always better than static for semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 67–73, Barcelona (online). International Committee for Computational Linguistics.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

John Philip McCrae. 2019. Identification of adjective-noun neologisms using pretrained language models. In *Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019)*, pages 135–141, Florence, Italy. Association for Computational Linguistics.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38:39–41.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Anna Mosolova, Marie Candito, and Carlos Ramisch. 2024. Injecting Wiktionary to improve token-level contextual representations using contrastive learning. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 34–41, St. Julian's, Malta. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. 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.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2020. Contrastive learning with hard negative samples. *ArXiv*, abs/2010.04592.

Joshua Robinson, Li Sun, Ke Yu, K. Batmanghelich, Stefanie Jegelka, and Suvrit Sra. 2021. Can contrastive learning avoid shortcut solutions? *Advances in neural information processing systems*, 34:4974–4986.

Dominik Schlechtweg, Anna Hätty, Marco Del Tredici, and Sabine Schulte im Walde. 2019. A wind of change: Detecting and evaluating lexical semantic change across times and domains. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 732–746, Florence, Italy. Association for Computational Linguistics.

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 Fourteenth Workshop on Semantic Evaluation*, pages 1–23, Barcelona (online). International Committee for Computational Linguistics.

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.

Ivan Vulić, Goran Glavaš, Fangyu Liu, Nigel Collier, Edoardo Maria Ponti, and Anna Korhonen. 2023. Probing cross-lingual lexical knowledge from multilingual sentence encoders. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2089–2105, Dubrovnik, Croatia. Association for Computational Linguistics.

Dong Wang, Ning Ding, Piji Li, and Haitao Zheng. 2021. CLINE: Contrastive learning with semantic negative examples for natural language understanding. 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 2332–2342, Online. Association for Computational Linguistics.

Wei Wang, Liangzhu Ge, Jingqiao Zhang, and Cheng Yang. 2022. Improving contrastive learning of sentence embeddings with case-augmented positives and retrieved negatives. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Lei Xu, Sarah Alnegheimish, Laure Berti-Equille, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2025. Single word change is all you need: Using llms to create synthetic training examples for text classifiers. *Expert Systems*, 42(8).

Hong Xuan, Abby Stylianou, Xiaotong Liu, and Robert Pless. 2020. Hard negative examples are hard, but useful. In *European Conference on Computer Vision*.

Sachin Yadav and Dominik Schlechtweg. 2025. Xl-durel: Finetuning sentence transformers for ordinal word-in-context classification. *ArXiv*, abs/2507.14578.

A Sentence-level collapse in token encoders for WiC

A natural concern about evaluating sense distinctions in contextualized embedding space is whether such representations encode lexical information at all, or whether they merely encode the sentential context surrounding a token. In our experiments with biencoders trained on WiC, we observe a pathology in which the encoder produces near-identical representations for distinct target tokens within the same sentence. To make this observation precise, and to connect it to a broader literature on representational anisotropy in contextualized models, we formalize the problem as follows.

Let $f : (S, i) \mapsto \mathbf{R}^d$ denote a token encoder, where $S = (t_1, \ldots, t_n)$ is a sentence and i is the index of a marked target token. A faithful lexical encoder should satisfy two minimal desiderata: that the representation vary as a function of the target position, since different tokens in the same sentence carry different lexical content; and that the variation across target positions within a sentence be commensurate with the variation across sentences for a fixed lexeme, since otherwise the representation cannot serve as the basis for any comparison of meaning at the word level. We argue that the encoders under study fail both desiderata, and we make this failure precise through an analytical formalization.

The encoder approximately factors through the sentence alone: there exists a sentence-level map $\varphi : S \mapsto \mathbf{R}^d$ and a residual term ε such that

$$f(S, i) = \varphi(S) + \varepsilon(S, i),$$

and we have the expectancy of

$$\mathbf{E}_{S,i} \|\varepsilon(S, i)\| \ll \mathbf{E}_{S \neq S'} \|\varphi(S) - \varphi(S')\|.$$

The contribution of the target index is dominated by the contribution of the sentence in which it is embedded. This is what we name **lexical insensitivity** in latent representations: up to small noise, $f(S, i) \approx f(S, j)$ whenever i, j index distinct positions within the same S.

B XL-LEXEME

Empirical observations in Figure 4 demonstrate that current state-of-the-art models prioritize context over individual word semantics. When testing synonyms and varied sentence structures, the model represents distinct words in identical contexts as more similar than identical senses in diverging contexts. Notably, similarity for identical senses often falls below the 0.75 threshold when context varies.

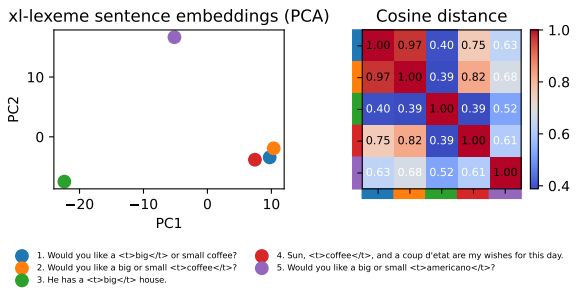

Figure 4: In left side we have a pc decomposition of the vectors produced by XL-LEXEME, on the right side we we have a cosine distance matrix between the sentences.

C Scheduling Adversarial Negatives

The early introduction of adversarial examples traps the model in a local optima, as we show in the Figure below.

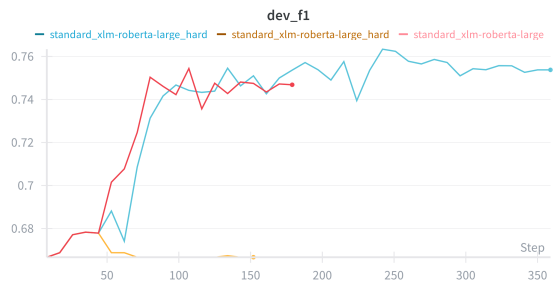

Figure 5: F1 score in the development data with different training schedules. Blue line show late insertion of LANE, yellow line insertion since first epoch, red line no insertion. Early stopping of 5 epochs.

The red line represent a training without adversarial examples, the yellow line the training with an early insertion of adversarial examples (first epoch), and the blue line the scheduled insertion of adversarial examples.

D Datasets Licensing and Usage

The datasets we compiled from existing resources (MASC, SEMCOR, FEWS), are under CC BY-SA 4.0. WIC and XL-WIC belongs to the original licensing CC BY-NC 4.0.

This data is intended to use for word in context differentiation, similar to the word-in-context tasks. In the table below we list the statistics of each dataset.

Dataset	Split	Instances
SEMCOR	Train	33,313
SEMCOR	Dev	7,000
SEMCOR	Test	9,674
MASC	Train	7,280
MASC	Dev	7,000
MASC	Test	2,968
FEWS	Train	132,237
FEWS	Dev	7,000
FEWS	Test	29,708
DWUG	Train	32,424
DWUG	Dev	7,000
DWUG	Test	6,993

Table 6: Dataset split sizes for SemCor, MASC, FEWS, and DWUG.

E Prompt for Annotating synonyms

Below we present the prompt to obtain in-sentence synonym annotation in JSON format. This prompt is used also in multi-lingual.

```
You are a linguistic analysis assistant.
Your task is to detect whether a sentence
contains two or more words that are
synonyms of each other.

A synonym pair must:
- Share the same or very similar meaning
in the context of the sentence
- Belong to the same part of speech (both
nouns, both verbs, etc.)
Synonyms can be:
- Morphological variants of the same root
(house and houses)
- Different capitalization of the same
word (Rift and rift)

Respond in JSON with this exact schema:
{
  "has_synonyms": true | false,
  "synonym_pairs": [
    {
      "word1": "<token>",
      "word2": "<token>",
      "shared_meaning": "<brief gloss>"
    }
  ],
  "reasoning": "<one sentence explaining
  your decision>"
}

If no synonym pairs are found, return an
empty list for "synonym_pairs". Output raw
JSON only, no markdown fences, no preamble.
```

F Compute Costs

The dynamic min-max candidate selection introduces computational overhead due to the additional forward passes required to evaluate candidate similarity in-batch. To train the models on SEMCOR, MASC, DWUG, WiC, and FEWS, we used an NVIDIA V100 32GB for approximately 40 hours. The XL-WiC training takes approximately 35 hours on the same hardware.

G Use of AI

We use AI as a code assistant and as a writing assistant, for improving grammar and readability.