In the LLM era, Word Sense Induction remains unsolved

Anna Mosolova[1,2], **Marie Candito**[1] , **Carlos Ramisch**[2]
[1]Université Paris Cité, CNRS, LLF, Paris, France
[2]Aix Marseille Univ, CNRS, LIS, Marseille, France
first.last@u-paris.fr, first.last@lis-lab.fr

Abstract

In the absence of sense-annotated data, word sense induction (WSI) is a compelling alternative to word sense disambiguation, particularly in low-resource or domain-specific settings. In this paper, we emphasize methodological problems in current WSI evaluation. We propose an evaluation on a SemCor-derived dataset, respecting the original corpus polysemy and frequency distributions. We assess pre-trained embeddings and clustering algorithms across parts of speech, and propose and evaluate an LLM-based WSI method for English. We evaluate data augmentation sources (LLM-generated, corpus and lexicon), and semi-supervised scenarios using Wiktionary for data augmentation, must-link constraints, number of clusters per lemma.

We find that no unsupervised method (whether ours or previous) surpasses the strong "one cluster per lemma" heuristic (1cpl). We also show that (i) results and best systems may vary across POS, (ii) LLMs have troubles performing this task, (iii) data augmentation is beneficial and (iv) capitalizing on Wiktionary does help. It surpasses previous SOTA system on our test set by 3.3%. WSI is not solved, and calls for a better articulation of lexicons and LLMs' lexical semantics capabilities.

1 Introduction

Disambiguating the senses of potentially ambiguous words in a text (i.e. word sense disambiguation, WSD) is a historic NLP task, essential for obtaining a formal representation of a text's meaning. However, this task has the drawbacks of (i) relying on predefined sense inventories of arbitrary granularity and ill-suited for specialized domains, and (ii) requiring labor intensive sense-annotated data, unavailable for most languages of the world. This requirement still holds in the Large Language Models (LLM) era: Sainz et al. (2023) and Anonymous (2025) show that open LLMs outperform BERT-based supervised systems only when fine-tuned on sense-labeled data.

The word sense induction task (WSI) does away with the need for a predefined sense inventory and sense-labeled data (except for evaluation), albeit at the expense of quality. In addition, one of the standard techniques in WSI is to cluster vector representations of a lemma's occurrences. When applied to all content words in a corpus, it provides corpus-dependent pseudo-sense labeling (Eyal et al., 2022). Although in the LLMs era the utility of induced senses is less clear for downstream applications, it remains central in computationally less intensive environments, corpus studies such as lexical change detection (Schlechtweg et al., 2020), or specific contexts such as scientific literature mining and sense-aware retrieval (Eyal et al., 2022).

Our work focuses on a "**full-corpus WSI scenario**", taking a corpus as input, inducing senses for all content-word lemmas occurring more than once (typically by clustering occurrences), yielding as a by-product pseudo-sense annotations. We note that popular WSI datasets created for SemEval shared tasks (Manandhar et al., 2010; Jurgens and Klapaftis, 2013) (i) exhibit artificial polysemy levels (because unambiguous lemmas are trivial to process, these datasets over-represent polysemous ones, thus not addressing the task of discovering which lemma is polysemous), (ii) and some exhibit artificial selection of lemmas and number of occurrences per lemma. State-of-the-art WSI systems for English are thus biased towards these unrealistic distributions, and it remains to prove that they work as well when faced with more natural data.

This paper makes the following contributions:

- We question the current evaluation in WSI: issues with standard datasets and metrics lead to methodological problems that hinder comparability and reproducibility (§ 3).

Findings of the Association for Computational Linguistics: ACL 2025, pages 17161–17178
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

- Building upon SemCor, we propose a more natural evaluation framework that respects the original polysemy and frequency distribution, and benchmark systems across datasets (§ 4).
- In this new framework, we assess clustering of pre-trained contextualized embeddings across parts of speech, and propose and evaluate direct LLM prompting for WSI for English (§ 5).
- We evaluate data augmentation sources (LLM-generated, corpus and lexicon), and semi-supervised scenarios using the English Wiktionary for data augmentation, must-link constraints, number of clusters per lemma (§ 6).

2 Related work

We focus here on previous work on WSI compatible with the full-corpus scenario (hence we ignore WSI approaches based on a lexical network such as Panchenko et al. (2017)). Such corpus-based approaches can be categorized into six groups:

(i) Clustering contextualized embeddings directly: the basic technique is to cluster contextualized embeddings produced by masked pre-trained language models (Devlin et al., 2019) (hereafter **PLM**). Liétard et al. (2024) perform two-step clustering[1] using BERT contextualized embeddings of target in-context words. The "local" step is a hard clustering of occurrences of a given lemma (the strict WSI task), while the "global" step agglomerates the centroids of the local clusters, hence obtaining clusters for "concepts" (equivalent to WordNet's synsets). Results on SemCor nouns (with more than 10 instances) show that the global step helps for the strict WSI task. The authors report a high non-weighted average F-B^3 (80%), but we point out that the average F-B^3 is usually *weighted* by each lemma's number of instances, which is all the more crucial in the full-corpus scenario.

(ii) Clustering contextualized embeddings enhanced for lexical semantics tasks: Abdine et al. (2022) train a small neural network to maximize the mutual information of pairs of original and perturbed instances. Then agglomerative clustering (AG) is used on vectors from the hidden layer of the network. AG is performed with either a fixed number of clusters or dynamically recomputed using a word polysemy quantification score (Xypolopoulos et al., 2021). The proposal of Yavas et al. (2024) also falls into this category. It consists in adversarial training of BERT to neutralize morphological and syntactic features, hypothesizing that they introduce noise for lexical semantics tasks. The authors perform K-means clustering of these modified contextualized embeddings, for all SemCor nouns and verbs, excluding those having a unique sense, and senses with less than 10 occurrences.

There are other works enhancing contextualized embeddings for lexical tasks, but not evaluating them for WSI. The main evaluation task for these is the Word-In-Context task (WiC, (Pilehvar and Camacho-Collados, 2019)), a binary classification task to decide whether two instances of the same lemma correspond to the same sense or not. In this vein, we can cite **MirrorWiC** (Liu et al., 2021) and the model of Mosolova et al. (2024) (which we will dub as **BERT-Wikt**). Both models were fine-tuned using contrastive learning, which teaches the model to bring semantically similar examples closer, while pushing dissimilar ones apart. MirrorWiC leverages self-supervised contrastive learning by using similar examples created by automatically alternating the original phrase. BERT-Wikt employs semi-supervised contrastive learning by using exemplars of the same sense from Wiktionary as similar ones.

(iii) Clustering vectors of BERT substitutes: In the Language-model Substitution with Dynamic Patterns model (**LSDP**), Amrami and Goldberg (2019) build vectors of BERT substitutes for each target instance, then cluster these vectors using agglomerative clustering. Hearst-like symmetric patterns are used to improve the quality of substitutes. Eyal et al. (2022) focus on scaling up this substitute-based technique, so that it can be used in the full-corpus scenario. The authors induce the senses of the 16k single-token words of the BERT-large (whole-word masking) vocabulary, and obtain a sense-labeled version of the English Wikipedia.

(iv) Learning sense embeddings using a masked language modeling objective: Ansell et al. (2021) propose the **PolyLM model**, which learns contextualized sense embeddings using a language modeling objective. For each lemma in the vocabulary, the model learns a fixed number of sense representations, and assigns in-context probabilities for each sense. The model builds on the assumptions that the probability of a word in context is the

[1] We note the similarity of this method with the semantic frame induction work of Yamada et al. (2021).

sum of the probabilities of all its senses and that for a given word occurrence, one of its senses should be more plausible than all the other ones. As a by-product, the model produces a probability distribution over senses for each word in context, which can be used to perform WSI. The authors report the SOTA results on SemEval 2010 and SemEval 2013 datasets.

(v) Latent-variable models: Amplayo et al. (2019) employ a latent variable model that models senses as distributions over multiple topics and uses target-neighbor pairs to induce more fine-grained senses and filter out the irrelevant ones.

(vi) WSI using LLMs: Larger decoder-only models have also been evaluated on lexical semantics tasks. Some have been shown to perform well for the WiC task (Hayashi, 2025). Ortega-Martín et al. (2023) report good ability of "ChatGPT" to identify ambiguity for specific words. Sumanathilaka et al. (2024) investigate LLMs' capabilities for WSD, using the English Wiktionary-derived FEWS dataset (Blevins et al., 2021). Results on a subset of the FEWS test set seem high but are unfortunately not compared to previous works. Sainz et al. (2023) and Anonymous (2025) show that open LLMs do outperform BERT-based supervised systems, but only when fine-tuned on sense-labeled data. Coming back to the WSI task, we can cite only one work involving LLMs: Periti et al. (2024) fine-tune LLMs on lexicographic definitions and exemplar sentences, for these LLMs to generate a definition given a word in context. The authors then perform WSI by clustering the embeddings of generated definitions, evaluated on a lexical semantic change dataset.

Importantly, these LLM approaches to WSD and WSI are computationally intensive, prompting the LLM once for each instance to disambiguate.

In the following, (i) to the best of our knowledge, we report the first results of directly asking the LLM to cluster sets of instances of a given lemma, a much more lightweight technique; (ii) we compare WSI performance across datasets, target lemma POS, and evaluation metrics for models falling into category (iii) (LSDP) and (iv) (PolyLM). The approaches (i) and (ii) are the focus of §5.

3 Issues in WSI evaluation

The evaluation of WSI models relies on datasets containing manual sense assignments for instances of a set of lemmas, and on metrics assessing how well the automatic assignment matches the manual one. Datasets have issues related to lemma and instance selection and dev/test data splits. They are often associated to heterogeneous and incomplete metrics, resulting in a complex landscape in which it is extremely difficult to compare, reproduce and/or replicate results.

3.1 Datasets

Two popular datasets for the evaluation of English WSI are those of SemEval 2010 Task 14 (Manandhar et al., 2010) and SemEval 2013 Task 13 (Jurgens and Klapaftis, 2013), used e.g. by Amrami and Goldberg (2019); Abdine et al. (2022); Eyal et al. (2022) (referred as SE10 and SE13 hereafter). See Appendix A for details on these and other datasets.

Pre-defined sense inventory evaluation bias Most WSI works evaluate systems using gold data labeled with senses from a pre-defined inventory. This introduces a bias since the granularity of sense distinction may vary across lexical resources and target objective tasks. Herman and Jakubíček (2024) proposed an evaluation dataset for Czech and English, later extended to 6 languages for the upcoming CoNLL 2025 shared task on robust WSI[2], designed to address this specific bias.

Senses distribution Both SE10 and SE13 datasets, as well as CoNLL 2025 robust WSI dataset, tend to over-represent polysemous and frequent lemmas. The first source of bias lies in the selection of lemmas, whose criteria are made explicit in neither of the datasets. Once lemmas are selected, the instances included in the test set come from OntoNotes v1.0 (SE10) and from the Open American National Corpus (SE13), but again the selection of these instances is not motivated. In SE13, all lemmas have at least 22 instances, with the majority having exactly 100. The CoNLL-25 WSI dataset focuses on 25 polysemous lemmas per language. As a result, WSI evaluation tends to ignore monosemous lemmas, albeit their high corpus frequency. Although polysemy detection models do exist, they are never applied in the context of WSI (Springorum et al., 2013; Lossio-Ventura et al., 2016; Habibi et al., 2021). In short, there is a mismatch between the sense distribution in these popular datasets and the more realistic full-corpus

[2] https://projects.sketchengine.eu/conll2025

WSI scenario.

Data splits While SE10 does not provide development set at all,[3] SE13 provides a trial dataset whose senses distribution is extremely different from the test set (different annotators, data sources, number of instances and polysemy levels). Participants who optimized their systems on this trial dataset obtained lower scores in the campaign (Jurgens and Klapaftis, 2013). More recent WSI work ignores the trial data, leading to a problematic use of the test set for hyperparameter tuning and comparison of configurations (Amrami and Goldberg, 2019; Abdine et al., 2022).

3.2 Metrics

Clustering is notoriously difficult to evaluate, with different metrics capturing different properties. In addition, most metrics are sensitive to sense distribution, questioning cross-dataset replicability.

Metrics heterogeneity Different evaluation metrics were used in SE10 (V-measure and Paired F-score) and SE13 (Fuzzy NMI and Fuzzy F-B^{3}[4]). The upcoming CoNLL 2025 shared task on robust WSI uses yet another metric based on rand index to take into account multiple gold annotations (Herman and Jakubíček, 2024). While previous work reporting results on SE datasets use the corresponding shared task metrics, works evaluated on other datasets use different metrics, making comparison all the more difficult. For instance, Yavas (2024) use adjusted rand index, Liétard et al. (2024) use F-B^3, Periti et al. (2024) use rand index, adjusted rand index, and purity, and Komninos and Manandhar (2016), following Li et al. (2014), adapt V-measure to use the best-upper-bound entropy estimator instead of maximum likelihood to alleviate some of its problems.

Metric properties Amigó et al. (2008) test the sensibility of metrics to four desirable properties: ***H****: cluster homogeneity* (clusters should not mix items belonging to different classes), ***C****: completeness* (items belonging to the same class should be grouped in the same cluster), ***RB****: rag bag* (adding an example of another class into clean cluster is worse than adding it into a mixed one) and ***SQ****: clusters size vs. quantity* (a small error in a big cluster should be preferable to a large number of small errors in small clusters). As shown in Table 1, F-B^3 is the only one sensitive to all four properties.

Metric	**H**	**C**	**RB**	**SQ**
Rand index	√	√	×	×
Paired F-score	√	√	×	×
NMI	√	×	×	√
V-measure	√	√	×	√
B^3 Precision	√	×	√	×
B^3 Recall	×	√	×	√
F-B^3	√	√	√	√

Table 1: Sensibility of clustering metrics to the properties defined by Amigó et al. (2008). Table 9 in Appendix B illustrates these properties on use cases.

Metric combination and comparison Works using the SE13 data report the geometric mean of fuzzy F-B^3 and fuzzy NMI (Amrami and Goldberg, 2019; Ansell et al., 2021). However, the latter is insensitive to completeness, artificially increasing when a class is divided into homogeneous clusters. Moreover, statistical significance is never reported, which weakens systems' comparison. Powerful significance tests for F-B^3 are computationally intensive because they require re-building the clusters for each bootstrapped resample. In our work, we test statistical significance for part of our experiments only (in §5).

For full-corpus WSI, it is crucial to use a metric counting instances (like F-B^3), compared to pair-based metrics, which overuse large gold classes, or compared to cluster-based metrics like NMI, which use proportions of gold classes in clusters, independently of their sizes. For these reasons, we adopt weighted average F-B^3 in our experiments (except in § 4, on cross-dataset variability).

4 Variation of systems' performance across datasets, metrics and POS

In this section, we investigate the impact of datasets' characteristics (polysemy level, target lemmas POS) and of evaluation metrics on performance in WSI.

We first describe our WSI evaluation dataset extracted from SemCor (Miller et al., 1993), respecting the original distributions of corpus occurrences and senses.

We then compare WSI performance across (i) recent state-of-the art systems (LSDP and PolyLM)[5],

[3] SE10's "training" data contains no annotation.

[4] The "fuzzy" versions are needed as SE13 asks soft clustering. Nonetheless, the SE10 metrics could have been adapted for soft clustering.

[5] We focused on the WSI previous works for which the code

and (ii) direct LLM prompting for WSI.

4.1 SemCor-WSI: extraction from SemCor

To overcome the flaws of the usual WSI datasets, we propose to tune and evaluate WSI systems on a dev and test sets extracted from SemCor 3.0 (Miller et al., 1993), a WordNet sense-annotated corpus[6]. We extract three sub-datasets of sense-annotated instances of verbs, nouns and adjectives: for each POS, we first consider the full lexicon of SemCor lemmas having this POS (including both single- and multiword lemmas) and occurring at least twice. Then for each POS, we (i) randomly select lemmas until we obtain approximately 10,000 corpus instances, and (ii) randomly split each POS's dev and test sets, keeping disjoint sets of lemmas, targeting the same number of instances, lemmas, and polysemy levels in both parts. We kept the first sense for instances annotated with multiple senses. So overall, the dataset contains both monosemous and polysemous lemmas, with instance counts per lemma varying from 2 to several hundreds, with a similar polysemy level in dev, test and full SemCor, as shown in Table 11 (Appendix E)[7].

Note that while this scenario addresses the unnaturalness of the SE10 and 2013 dataset pointed in §3, it does not address the pre-defined sense inventory evaluation bias (addressed by Herman and Jakubíček (2024) and in the upcoming CoNLL 2025 shared task on robust WSI). We leave it for future work to combine a scenario evaluating both on a natural sense distribution and circumventing the pre-defined senses' bias.

4.2 Experimental protocol

We evaluate five models, starting with the state-of-the-art PolyLM model (Ansell et al., 2021).We compare PolyLM base (54M parameters) and PolyLM large (90M parameters). Another model is LSDP by Amrami and Goldberg (2019)[8]. We report the average and standard deviation of 10 runs as per the authors' methodology. To adapt LSDP for the SemCor-WSI dataset, we modified the process of determining strong and weak senses (see Appendix D for details). We reuse the default hyperparameters set by the authors.

We also test 3 large language models: the proprietary GPT 4-o (OpenAI et al., 2024) and two open-source models: Llama 3.1 8B Instruct (Grattafiori et al., 2024) and Llama 3.3 70B Instruct (4 bit). For each model, we use identical prompts adapted to the specific task (full prompts, prompt tuning details and exact LLMs versions are given in Appendix J). We report both the average and standard deviation of five runs for each LLM.

For each dataset, we provide two simple baselines: one cluster per lemma (1cpl) and one cluster per example (1cpex).

4.3 Results and discussion

Table 2 shows the performance of each model (results per POS provided in Appendix M). For SE10, we provide also the non-fuzzy versions of the SE13 metrics, and for comparison, we use these for SemCor-WSI (F-B^3 and NMI). Globally, PolyLM-large is the best-performing model having the highest results for 4 metrics across three datasets (but this does not hold for all individual POS). **None of the models surpasses the high 1cpl baseline in F-B^3 on SemCor-WSI**, a result likely to hold for full-corpus scenario on other corpora.

The evaluation metrics do not always follow the same patterns. On SE10, the best model differs for each of the four reported metrics. For SE13 and SemCor-WSI, F-B^3 and NMI (and their fuzzy versions) correlate better.

Moreover, on F-B^3, the 1cpl baseline performs best for SE10 and SemCor-WSI. For SE10, it suggests that optimizing the systems for the original metrics is detrimental when changing metrics.

Performance of LLMs is globally much lower (except on verbs, see Table 17 in Appendix), with sometimes huge variance. Llama models struggle with processing lemmas with large numbers of examples (present in SE10, and in the full-corpus scenario). They tend to forget the task after 300 examples and either halt generation, repeat the same answer, or produce irrelevant sentences[9]. While the GPT-4o model does not suffer from this limitation, all LLMs occasionally produce unnecessary explanations, ask for a sense inventory, or simply "refuse" to perform the task. Moreover, parsing answers revealed difficult, in particular for sense

is available and functional, or for which there are reported results on Semeval 2010 and/or 2013 WSI datasets.

[6] We used the brown1 and brown2 parts, which are the only ones having sense annotations for all open class words.

[7] The code and the dataset are available at: https://github.com/anya-bel/fullcorpus_wsi

[8] https://github.com/asafamr/BERTwsi

[9] Processing sets of instances of a given lemma in batches should be investigated, but requires to merge the sense inventory induced for each batch.

Model	SemEval 2013		SemEval 2010				SemCor-WSI	
	Fuzzy-NMI	Fuzzy-F-B^3	V-M	Paired F-S	NMI	F-B^3	F-B^3	NMI
PolyLM large	**23.7**	**66.7**	**43.6**	67.5	6.2	49.2	73.0	**33.6**
PolyLM base	23.0	65.4	41.8	66.4	6.2	49.1	71.3	31.5
LSDP	21.1[±0.6]	64.1[±0.5]	38.9[±1.0]	**70.7**[±0.4]	4.6[±0.1]	52.8[±0.2]	71.0[±0.4]	32.1[±0.7]
Llama 3.1 8B	2.3[±0.4]	57.1[±0.5]	16.5[±0.9]	49.3[±1.2]	7.3[±0.5]	49.6[±0.9]	59.7[±1.0]	19.4[±0.8]
Llama 3.3 70B	8.9[±0.4]	44.2[±1.8]	29.4[±0.9]	49.7[±4.8]	8.1[±0.6]	49.6[±1.6]	64.2[±0.9]	27.8[±1.1]
GPT-4o	16.9[±0.5]	58.6[±1.6]	36.3[±2.0]	63.9[±2.0]	7.1[±0.3]	47.7[±1.9]	66.9[±0.7]	29.2[±1.2]
1cpl	0.0	61.23	0.0	63.5	0.0	**64.1**	**73.6**	28.1
1cpex	6.9	NA	31.7	0	**19.5**	8.0	24.1	20.7

Table 2: WSI results on 3 datasets. PolyLM large/base: for SE10/SE13, results reproduced using models of https://github.com/AlanAnsell/PolyLM, for SemCor-WSI, results obtained using their code. LSPD: substitutes obtained with BERT-large. For SE10/SE13, results reproduced with the code of Amrami and Goldberg (2019), for SemCor-WSI, results obtained using the adapted code (see text).

applicability degrees in SE13.Note though that despite these flaws, LLMs sometimes produced interpretable cluster names, which is not straightforward with traditional approaches.

5 Investigating full-corpus WSI

We reported earlier that the top-performing systems do not surpass the 1cpl baseline when switching to a more naturally distributed dataset. This suggests that over-representation of polysemy in earlier datasets may have influenced the systems' design. In this section, we investigate the performance achievable on SemCor-WSI using the basic technique of clustering the contextualized embeddings of a given lemma instances and we hypothesize that performance may vary depending on polysemy injecting lexical information either using unlabeled . We perform a grid search using (i) two clustering algorithms which automatically determine the number of clusters (X-Means and AG$_{silh}$), (ii) two BERT PLMs, plus PLMs fine-tuned to better perform on WiC task. This allows us to assess which model and algorithm combination performs best on a more naturally distributed dataset.

5.1 Experimental protocol

Contextualized embeddings We test the *base-uncased* and *large-uncased* versions of BERT (Devlin et al., 2019)[10] (BERT-b-u and BERT-l-u hereafter). We also test two models fine-tuned for the WiC task, likely to benefit for WSI: *MirrorWiC-base* (Liu et al., 2021) and *BERT-Wikt* (Mosolova et al., 2024). [11]. For the latter, we ran the fine-tuning procedure on all POS with default hyperparameters on BERT-l-u to obtain *BERT-l-Wikt* model. For each PLM, we tested all layers and report using the best-performing layer (see Appendix G).

Clustering algorithms We test AG clustering with silhouette score (AG$_{silh}$) to determine the optimal number of clusters, and X-means, which dynamically determines the number of clusters. Being based on K-means with K++ initialization, X-means is not deterministic, we thus report the average and standard deviation of 5 runs. Hyperparameters are provided in Appendix L, including default number of clusters when silhouette is not defined.

Handling of POS variation Sense distribution varies across parts of speech (cf. Table 11). To study the impact of these differences, we provide results per POS and for all POS (**All POS**). Moreover, the number of lemmas for each POS is almost balanced in our subsets, but not in the full SemCor. So we also show results averaged over the 3 POS weighted by their proportions in the full SemCor ($_w$**Avg**) to reflect their natural distribution in corpus (the proportions are provided in Appendix F).

Metrics and Statistical significance We use F-B^3, as motivated in §3. Due to computational costs, we chose to perform the bootstrapping statistical significance test for all PLM pairs combined with AG$_{silh}$ only and not X-means (5 reruns are needed for the latter, see details in Appendix H).

5.2 Results and discussion

Results are shown in Table 3[12].

[10] For all experiments with PLMs, we use Transformers library (Wolf et al., 2020). Subword embeddings are averaged per word or MWE.

[11] For all our experiments, we use the dbnary dump of 06/12/2024, https://kaiko.getalp.org/about-dbnary/. As in (Mosolova et al., 2024), 20% is not used, to keep the possibility to evaluate on unused Wiktionary data.

[12] Statistical significance tests (using AG$_{silh}$) show that differences between all pairs of PLMs are significant at p < 0.05, except: i) All POS: BERT-l-u versus MirrorWiC , ii)

Model	Algo	Verb	Adj	Noun	All POS	${}_{w}$Avg
Unsupervised						
BERT-b-u	AG$_{silh}$	64.8	75.6	72.3	70.6	70.8
	X-Means	62.9[±0.1]	**76.5**[±2.2]	73.7[±0.3]	70.6[±0.8]	71.1
BERT-l-u	AG$_{silh}$	**65.8**	75.7	72.3	**71.1**	71.1
	X-Means	63.2[±0.6]	75.5[±1.2]	**74.8**[±0.1]	70.2[±0.1]	**71.5**
Self-supervised						
MirrorWiC-base	AG$_{silh}$	**65.1**	74.7	70.9	70.2	70.0
	X-Means	63.0[±0.2]	**77.0**[±1.5]	**74.4**[±0.2]	**71.2**[±0.4]	**71.6**
Semi-supervised						
BERT-l-Wikt	AG$_{silh}$	**67.8**	**75.5**	72.4	**71.8**	**71.7**
	X-Means	63.9[±1.3]	74.9[±1.7]	**74.5**[±0.5]	69.7[±0.3]	71.5
Baselines	1cpl	65.7	**80.0**	**75.2**	**73.6**	**73.4**
	1cpex	25.5	22.6	24.1	24.1	24.2

Table 3: F-B^3 performance across PLMs and clustering algorithms for each POS, for all POS (All POS), and the average over POS weighted by POS proportion in SemCor (${}_{w}$Avg). In bold, the best value for each model supervision type (un-, self-, semi-supervised). In blue, the best value for each column, excluding baselines. In red, cases where a baseline is best over the column. Best previous system: PolyLM-large: 73.0 (Table 2).

Baselines are high: The first striking observation is that the 1 cluster per lemma "baseline" is actually the best technique for adjectives and nouns, and when considering all POS (All POS: 73.6, ${}_{w}$Avg: 73.4). The other systems only surpass 1cpl for verbs, namely for the most polysemous POS.

Models: Among unsupervised embeddings models, BERT-l-u outperforms its base counterparts, overall, except for adjectives.The self-supervised finetuning of MirrorWiC surpasses the unsupervised BERTs for adjectives, but not for nouns and verbs, giving a marginal improvement overall.

The semi-supervised models (fine-tuned for WiC on Wiktionary) provide the best performance (excluding baselines), both for verbs and for all POS.

Variation across POS: The results show that the tendencies across POS vary greatly. Using contextualized embeddings fine-tuned on Wiktionary does help in general, but not for adjectives, for which the 1cpl and then unsupervised models perform best. The tendency is opposite for verbs.

Clustering algorithms: AG$_{silh}$ performs always better for verbs, while X-means performs always better for nouns, and most of the time for adjectives. This could be explained by X-means' tendency to define less clusters, which is beneficial for POS with lower polysemy rate. For the results over the 3 POS (All POS and ${}_{w}$Avg) X-means tends to outperform AG$_{silh}$, except for the semi-supervised models. Across model/algorithms, the best pair is semi-supervised model plus AG$_{silh}$.

Taking these observations into account, and considering that X-means is not deterministic and needs to be run several times, we will use AG$_{silh}$ in the following experiments, the best-performing unsupervised model (BERT-l-u) and the semi-supervised BERT-l-Wikt model.

So for now, on a dataset more natural in terms of polysemy and sense distribution, these contextualized embeddings clustering techniques do not surpass the best previous system (PolyLM: 73.0), and none surpasses the 1cpl technique (73.6).

6 Investigating data augmentation

In this section, we investigate the simple technique of adding unlabeled examples to the set of instances to cluster. Augmenting the set of instances makes it denser, potentially creating new similarity links, in particular for lemmas with originally few instances. Moreover in such cases, considering more instances helps to avoid undefined silhouette cases, defaulting to one cluster (cf. Appendix L).

We investigate 1) unsupervised augmentation, adding either attested examples from external corpora or synthetic examples generated by LLMs; 2) semi-supervised augmentation, where we leverage Wiktionary examples for either direct dataset augmentation (based on the Wiktionary senses),

Nouns: BERT-b-u versus MirrorWiC and iii) Verbs: BERT-l-Wikt versus MirrorWiC. See also Fig. 1 in Appendix.

and/or fine-tuning the embedding model (BERT-l-Wikt model (Mosolova et al., 2024) already used in previous sections).

6.1 Dataset augmentation

For each lemma, we augment the set of instances with either (i) Wikibooks[13] instances, (ii) LLM-generated sentences, and (iii) Wiktionary exemplar sentences (independently of their senses).

For Wikibooks (WB), we extract all occurrences of all SemCor-WSI nouns, verbs and adjectives. For each lemma, we randomly select at most N examples (N=10, 50, 100, 150). For Multiword lemmas (MWEs) examples are retrieved based on the first word of the MWE. We handle MWEs in the same way to retrieve the Wiktionary instances.

We also test LLM-generated data.For each instance in our dataset, we provide it to the model which we prompt to generate 3 examples with same sense (the exact prompt is provided in Appendix K). We use a small open-source Llama 3.1 8B 4bit and a proprietary GPT-4o[14]. Appendix I provides the total number of added examples in each setting.

6.2 Constrained clustering

We extend the AG clustering algorithm by incorporating must-link constraints. For each lemma, we add all Wiktionary examples and impose must-link constraints between examples assigned to the same Wiktionary sense (assigning distance 0 for all such pairs). For each lemma's number of clusters, we either use Wiktionnary's number of senses ($\mathbf{AG}_{wikt}$), or the silhouette score ($\mathbf{AG}_{silh}$).

6.3 Results and discussion

We conducted experiments with BERT-l-u and BERT-l-Wikt embedding models (Tables 4 and 5).

We observe the regular trend that whatever embedding model and data augmentation source, adding examples systematically improves results.

A new SOTA technique: The previous SOTA system (PolyLM) requires training a masked language model from scratch. Yet, Table 4 shows that it can be outperformed simply by adding sufficient unlabeled data during the clustering of contextualized embeddings: all the settings using at least 50 WikiBooks instances do surpass the PolyLM's performance (73.0 in Table 2)[15].

Additional examples: Comparing sources of examples, adding 10 or more WB examples per lemma results in better performance than adding Wiktionary examples, although corresponding to a similar number of examples (the exact numbers provided in Appendix I). On the contrary, adding more than 45k LLM-generated examples is comparable to the WB 10 examples per lemma setting. Moreover, adding more and more WB examples helps, up to the limit of 150 examples per lemma. So overall, adding attested corpus examples helps more than the litterary style examples from Wiktionary, and more than the LLM-generated examples. To conclude, adding around 100/150 examples from raw corpora per lemma is both the cheapest and the best option.

Must-link constraints: Concerning must-link constraints, we can observe that AG$_{silh}$ with must-link is very slightly better than without (columns 1 and 3 of Tables 4 and 5), except for the Wiktionary data augmentation: in the former case, we add Wiktionary examples to the other augmentation source. It seems that the improvement of must-link per se only stems from the addition of more examples.

Number of clusters: Comparing columns 2 and 3 of Tables 4 and 5, shows that using the Wiktionary number of clusters is systematically better than silhouette (even if Wiktionary's sense inventory differs from WordNet's).

BERT-large vs BERT-l-Wikt: Using the fine-tuned BERT model (Table 5) is always better than the corresponding BERT-large model (Table 4).

Comparison to the 1cpl baseline: Finally, several settings do outperform the 1cpl baseline (73.6, surpassing results are shown in blue and red in Tables 4 and 5), ***but only for settings using Wiktionary in some way***. The best result (75.7) uses it in three ways (in the embedding model, the must-link constraints which also adds the Wiktionary examples, and to define the number of clusters). While this method does use a lot of manually annotated data (a full Wiktionary), we would like to emphasize that it is usable for the many languages for which a large Wiktionary exists[16].

[13] https://huggingface.co/datasets/bigscience-data/roots_en_wikibooks

[14] model name: gpt-4o-2024-08-06, generation seed: 42

[15] Note that PolyLM cannot benefit from data augmentation, unless by retraining a full sense embeddings model.

[16] 19 languages have Wiktionary with more than 100k entries.

Augmentation	**Base**	**Must-link**	
source	**AG**$_{silh}$	**AG**$_{wikt}$	**AG**$_{silh}$
None	71.1$^{\diamond}$	NA	NA
Wiktionary	72.1	71.9	72.2
Llama 3.1 8B 4bit	71.9$^{\diamond}$	73.6	72.7
GPT-4o	72.4$^{\diamond}$	73.6	72.9
WB (10 per l.)	72.6$^{\diamond}$	73.5	73.0
WB (50 per l.)	73.5$^{\diamond}$	74.1	**73.4**
WB (100 per l.)	73.5$^{\diamond}$	74.2	73.2
WB (150 per l.)	**73.6**$^{\diamond}$	**74.4**	73.3

Table 4: F-B^3 results for All POS: AG clustering using BERT-l-u embeddings for various augmentation sources, with or without must-link constraints, using the nb of clusters from silhouette (**AG**$_{silh}$) or from Wiktionary (**AG**$_{wikt}$). $^{\diamond}$ indicates unsupervised results, all the other ones using Wiktionary in some way. Results above 1cpl (73.6, cf. Table 3) are in blue.

Augmentation	**Base**	**Must-link**	
source	**AG**$_{silh}$	**AG**$_{wikt}$	**AG**$_{silh}$
None	71.8	NA	NA
Wiktionary	72.7	74.1	72.4
Llama 3.1 8B 4bit	72.3	74.7	73.7
GPT-4o	72.9	75.1	73.8
WB (10 per l.)	73.6	75.2	73.7
WB (50 per l.)	73.9	75.3	**73.8**
WB (100 per l.)	**74.5**	**75.7**	73.8
WB (150 per l.)	74.3	**75.7**	73.8

Table 5: Same as Table 4 but using BERT-l-Wikt. Results above 1cpl (73.6, cf. Table 3) are in blue.

Model	**F-B**^{3}
PolyLM-large	72.7
BERT-l-uncased+AG$_s$+150WB	74.0
BERT-l-Wikt+AG$_{wikt}$+ML+100WB	**76.0**
1cpl	74.4

Table 6: F-B^3 results on SemCor-WSI test set for all POS of the best previous system (cf. Table 2) and the best unsupervised and supervised models from Tables 4 and 5 (for these models we reuse the best layer tuned on the development set).

Results on test set We check if the observed trends are confirmed on the test set, comparing performance for 1cpl, the best previous system (PolyLM-large), our best unsupervised system (BERT-l-u-AG$_s$+150WB) and our best Wiktionary-using system (BERT-l-Wikt-MustLink-AG$_{wikt}$+100WB). Results in Table 6 show that the Wiktionary-enhanced system (76.0) is best, followed by 1cpl (74.4), our unsupervised system (74.0) and previous SOTA system PolyLM (72.7).

7 Conclusions

In this paper, we advocated for evaluating WSI on data respecting more natural distributions of occurrences and number of senses per lemma. We proposed experiments on English, evaluated on an extract of SemCor, both reusing state-of-the-art previous methods, and investigating an LLM prompting technique, data augmentation, and a semi-supervised setting where the English Wiktionary is used in three ways (as a source for fine-tuning a BERT model on the Word-In-Context task (Mosolova et al., 2024), for data augmentation and for must-link clustering constraints).

The LLM prompting technique we proposed lays far behind, calling for better leveraging the lexical semantics knowledge of LLMs.

Our striking conclusion is that, when considering a dataset following a more natural sense distribution and polysemy level, none of the fully unsupervised systems we tested surpass the simple baseline of clustering all instances of the same lemma into a single cluster (considering a dataset with an average polysemy close to 2, cf. Table 11).

Simple data augmentation allows to surpass the much more complex previous SOTA model (PolyLM). BERT embeddings fine-tuned using contrastive learning on Wiktionary examples (Mosolova et al., 2024) are always better compared to the original BERT embeddings. More generally, we showed several ways to leverage Wiktionary allowing to surpass 1cpl on the dev and test sets (must-link constraints, data augmentation, definition of the number of clusters, fine-tuning of PLMs). Note this can be applied for the languages having a large Wiktionary or another electronic lexicon.

8 Future Work

We report mediocre performance, when directly prompting LLM for the WSI task. However, several techniques could be explored to improve LLM performance: (i) usage of chain-of-thought prompting to first ask for a list of senses, and then to assign the instances to each of the LLM-induced senses; (ii) processing sets of instances of a given lemma in batches, followed by a merging procedure for the sense inventories induced for each batch.

Limitations

This paper provides an overview of the existing datasets for Word Sense Induction, introduces a new evaluation framework for this task, and uses it to establish baselines and test data augmentation techniques to improve baseline results. However, our study evaluates only three large language models and four pre-trained language models in combination with two clustering algorithms. Additionally, this research is conducted in the English language and uses only a small part of SemCor.

Despite reporting the mean and standard deviation of 5 runs for non-deterministic models, we provide the statistical significance for result differences only for some deterministic models. Since powerful statistical significance tests for F-B^3 involve using bootstrapped resampling, running 1000 iterations for all configurations discussed in this paper would require prohibitively expensive computational resources.

This study was also limited by available GPU resources which included a single Nvidia A100 80GB GPU. Thus, we could not report results on the full Llama 3.3 70B model (which requires at least 135GB of GPU memory), and instead used its quantized version.

We should have included results on adverbs as well and plan to do so for a more complete evaluation.

Acknowledgements

We thank the reviewers for their valuable feedback on our work.

This work has been funded by the French Agence Nationale pour la Recherche, through the SELEXINI project (ANR-21-CE23-0033-01).

References

Hadi Abdine, Moussa Kamal Eddine, Michalis Vazirgiannis, and Davide Buscaldi. 2022. Word sense induction with hierarchical clustering and mutual information maximization.

Eneko Agirre and Aitor Soroa. 2007. SemEval-2007 task 02: Evaluating word sense induction and discrimination systems. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 7–12, Prague, Czech Republic. Association for Computational Linguistics.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2008. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12(4):461–486.

Reinald Kim Amplayo, Seung-won Hwang, and Min Song. 2019. Autosense model for word sense induction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6212–6219.

Asaf Amrami and Yoav Goldberg. 2019. Towards better substitution-based word sense induction. *Preprint*, arXiv:1905.12598.

Anonymous. 2025. Are large language models good word sense disambiguators? In *Submitted to ACL Rolling Review - December 2024*. Under review.

Alan Ansell, Felipe Bravo-Marquez, and Bernhard Pfahringer. 2021. PolyLM: Learning about polysemy through language modeling. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 563–574, Online. Association for Computational Linguistics.

Chris Biemann. 2012. Turk bootstrap word sense inventory 2.0: A large-scale resource for lexical substitution. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 4038–4042, Istanbul, Turkey. European Language Resources Association (ELRA).

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.

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

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.

Rotem Dror, Lotem Peled-Cohen, Segev Shlomov, and Roi Reichart. 2020. *Statistical Significance Testing for Natural Language Processing*. Springer International Publishing.

Katrin Erk and Diana McCarthy. 2009. Graded word sense assignment. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 440–449, Singapore. Association for Computational Linguistics.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. 2009. Investigations on word senses and word usages. In *Proceedings of the Joint Conference of the 47th*

Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, pages 10–18, Suntec, Singapore. Association for Computational Linguistics.

Matan Eyal, Shoval Sadde, Hillel Taub-Tabib, and Yoav Goldberg. 2022. Large scale substitution-based word sense induction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4738–4752, Dublin, Ireland. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Amir Ahmad Habibi, Bradley Hauer, and Grzegorz Kondrak. 2021. Homonymy and polysemy detection with multilingual information. In *Proceedings of the 11th Global Wordnet Conference*, pages 26–35, University of South Africa (UNISA). Global Wordnet Association.

Yoshihiko Hayashi. 2025. Evaluating LLMs' capability to identify lexical semantic equivalence: Probing with the word-in-context task. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 6985–6998, Abu Dhabi, UAE. Association for Computational Linguistics.

Ondřej Herman and Miloš Jakubíček. 2024. ShadowSense: A multi-annotated dataset for evaluating word sense induction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14763–14769, Torino, Italia. ELRA and ICCL.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA. Association for Computational Linguistics.

Nancy Ide and Keith Suderman. 2004. The American national corpus first release. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

David Jurgens and Ioannis Klapaftis. 2013. SemEval-2013 task 13: Word sense induction for graded and non-graded senses. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 290–299, Atlanta, Georgia, USA. Association for Computational Linguistics.

Alexandros Komninos and Suresh Manandhar. 2016. Structured generative models of continuous features for word sense induction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3577–3587, Osaka, Japan. The COLING 2016 Organizing Committee.

Linlin Li, Ivan Titov, and Caroline Sporleder. 2014. Improved estimation of entropy for evaluation of word sense induction. *Computational Linguistics*, 40(3):671–685.

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.

Qianchu Liu, Fangyu Liu, Nigel Collier, Anna Korhonen, and Ivan Vulić. 2021. MirrorWiC: On eliciting word-in-context representations from pretrained language models. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 562–574, Online. Association for Computational Linguistics.

Juan Antonio Lossio-Ventura, Clement Jonquet, Mathieu Roche, and Maguelonne Teisseire. 2016. Automatic biomedical term polysemy detection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1684–1688, Portorož, Slovenia. European Language Resources Association (ELRA).

Suresh Manandhar, Ioannis Klapaftis, Dmitriy Dligach, and Sameer Pradhan. 2010. SemEval-2010 task 14: Word sense induction &disambiguation. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 63–68, Uppsala, Sweden. Association for Computational Linguistics.

Rada Mihalcea, Timothy Chklovski, and Adam Kilgarriff. 2004. The senseval-3 English lexical sample task. In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 25–28, Barcelona, Spain. Association for Computational Linguistics.

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.

Roberto Navigli and Daniele Vannella. 2013. SemEval-2013 task 11: Word sense induction and disambiguation within an end-user application. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 193–201, Atlanta, Georgia, USA. Association for Computational Linguistics.

Andrei Novikov. 2019. Pyclustering: Data mining library. *Journal of Open Source Software*, 4(36):1230.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Miguel Ortega-Martín, Óscar García-Sierra, Alfonso Ardoiz, Jorge Álvarez, Juan Carlos Armenteros, and Adrián Alonso. 2023. Linguistic ambiguity analysis in chatgpt. *Preprint*, arXiv:2302.06426.

Alexander Panchenko, Eugen Ruppert, Stefano Faralli, Simone Paolo Ponzetto, and Chris Biemann. 2017. Unsupervised does not mean uninterpretable: The case for word sense induction and disambiguation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 86–98, Valencia, Spain. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Francesco Periti, David Alfter, and Nina Tahmasebi. 2024. Automatically generated definitions and their utility for modeling word meaning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14008–14026, Miami, Florida, USA. Association for Computational Linguistics.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations. *arXiv preprint*. ArXiv:1808.09121 [cs].

Oscar Sainz, Oier Lopez de Lacalle, Eneko Agirre, and German Rigau. 2023. What do language models know about word senses? zero-shot WSD with language models and domain inventories. In *Proceedings of the 12th Global Wordnet Conference*, pages 331–342, University of the Basque Country, Donostia - San Sebastian, Basque Country. Global Wordnet Association.

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.

Sylvia Springorum, Sabine Schulte im Walde, and Jason Utt. 2013. Detecting polysemy in hard and soft cluster analyses of German preposition vector spaces. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 632–640, Nagoya, Japan. Asian Federation of Natural Language Processing.

T. G. D. K. Sumanathilaka, Nicholas Micallef, and Julian Hough. 2024. Can llms assist with ambiguity? a quantitative evaluation of various large language models on word sense disambiguation. *Preprint*, arXiv:2411.18337.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Christos Xypolopoulos, Antoine Tixier, and Michalis Vazirgiannis. 2021. Unsupervised word polysemy quantification with multiresolution grids of contextual embeddings. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3391–3401, Online. Association for Computational Linguistics.

Kosuke Yamada, Ryohei Sasano, and Koichi Takeda. 2021. Semantic frame induction using masked word embeddings and two-step clustering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 811–816, Online. Association for Computational Linguistics.

Deniz Ekin Yavas. 2024. Assessing the significance of encoded information in contextualized representations to word sense disambiguation. In *Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language*, pages 42–53, Malta. Association for Computational Linguistics.

Deniz Ekin Yavas, Timothée Bernard, Laura Kallmeyer, and Benoît Crabbé. 2024. Improving word sense induction through adversarial forgetting of morphosyntactic information. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 238–251, Mexico City, Mexico. Association for Computational Linguistics.

A WSI datasets

A.1 SemEval 2010 Task 14

Manandhar et al. (2010) propose a WSI task, the goal of which is to train a system on an unannotated corpus and then use it to annotate test instances of the lemmas present in the training corpus. The dataset contains instances of 100 lemmas. The unsupervised training set is composed of contexts for each lemma obtained through automated queries using WordNet-related word lemmas. The test set contains unseen instances of each lemma originating from OntoNotes (Hovy et al., 2006) annotated with OntoNotes senses. We note that, at the time, OntoNotes had an annotation only of the most frequent polysemous lemmas within a subset of PropBank. Statistics on the dataset size are given in Table 7.

.

	Training set	Testing set	Senses (AVG)
All	879807	8915	3.79
Nouns	716945	5285	4.46
Verbs	162862	3630	3.12

Table 7: Training & testing set details from SemEval 2010 paper (Manandhar et al., 2010).

A.2 SemEval 2013 Task 13

Jurgens and Klapaftis (2013) introduced a new task which consists in annotating instances of lemmas with one or more senses and weighting each by their applicability (Graded Word Sense Induction). The dataset is divided into two parts: a trial set and a testing set. The trial set includes 8 lemmas, each with 50 contexts (data gathered by Erk and McCarthy (2009)). The test set contains 50 lemmas, with each lemma having between 22 and 100 contexts, annotated using WordNet senses. An important consideration for this dataset is the nature and the annotation difference for the trial and testing sets. The trial set is composed of a mix of 25 SemCor (Miller et al., 1993) and 25 SENSEVAL-3 (Mihalcea et al., 2004) random examples, while the test set was gathered from the Open American National Corpus (Ide and Suderman, 2004). The annotation process of the trial set was performed by three untrained lexicographers who evaluated the applicability of each WordNet sense on a 5 point scale (Erk et al., 2009), while the testing set was annotated by the authors of the paper (Jurgens and Klapaftis, 2013) on a 4 point scale. The dataset statistics are presented in Table 8.

	Testing set	Trial set
Instances	4664	400
AVG senses/inst.	1.12	4.97

Table 8: Testing set details from SemEval 2013 paper (Jurgens and Klapaftis, 2013), trial set details computed on the provided dataset. **AVG senses/inst.**: mean number of applicable senses per instance.

A.3 Other WSI Datasets

Other datasets for WSI evaluation include SemEval 2007 Task 2 (Agirre and Soroa, 2007), which is replaced by SemEval 2010 (and has similar issues), SemEval 2013 Task 11 (Navigli and Vannella, 2013) on clustering web query results, and corresponding to WSI when queries contain single words, and the aforementioned CoNLL-2025 Robust WSI Task, whose final evaluation data was not yet released.

Beyond shared tasks, some authors of WSI models proposed their own datasets and metrics because of the issues discussed in Section 3. Eyal et al. (2022) create their own dataset by annotating 20 ambiguous lemmas each represented by 100 random contexts from English Wikipedia to evaluate their large scale system across sense-induced Wikipedia. This dataset is not available online, therefore we did not report results on it. Panchenko et al. (2017) used SemEval 2013 Task 13 and the dataset proposed by Biemann (2012) to evaluate different configurations of their system. This dataset later was not used by other authors, it is also out of scope for our paper.

B Evaluation metrics properties

Table 9 presents our replication of the analysis by Amigó et al. (2008), including WSI simple baselines: one cluster per lemma and one clsuer per instance.

C GPU usage

For all experiments described in this paper, we used a single Nvidia A100 80GB GPU card. In Table C, we detail the GPU memory requirements and processing times for each model when using the SemCor-WSI dataset. The total GPU computational time for all experiments reported in this paper is 67.5 hours, excluding the time spent on

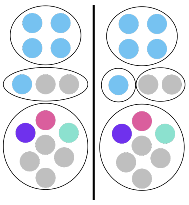
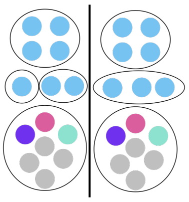
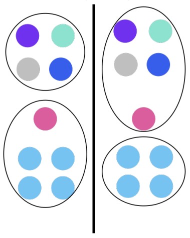
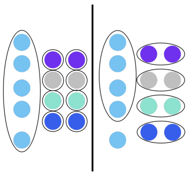
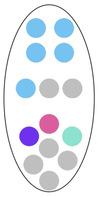
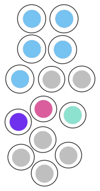
Metric	Homogeneity			Completeness			Rag Bag			Size vs Quality			1cpl	1cpex
														
Rand index	0.68	0.7	√	0.68	0.7	√	0.72	0.72	×	0.95	0.95	×	0.27	0.73
Paired F-score	0.47	0.49	√	0.47	0.53	√	0.55	0.55	×	0.83	0.83	×	0.43	ND
NMI	0.45	0.56	√	0.55	0.55	×	0.43	0.43	×	0.78	0.89	√	0.0	0.49
V-measure	0.5	0.58	√	0.57	0.6	√	0.61	0.61	×	0.88	0.94	√	0.0	0.66
B^3 Precision	0.6	0.69	√	0.69	0.69	×	0.49	0.56	√	1.0	1.0	×	0.33	1.0
B^3 Recall	0.7	0.7	×	0.71	0.76	√	1.0	1.0	×	0.69	0.88	√	1.0	0.36
F-B^3	0.64	0.69	√	0.7	0.72	√	0.66	0.71	√	0.82	0.93	√	0.49	0.53

Table 9: Verification of clustering metrics properties proposed by Amigó et al. (2008) on their use cases (see figure 11 in their paper) for the metrics discussed in the §3. We also test the metrics on the 1 cluster per lemma and 1 cluster per instance solutions.

statistical significance tests. The experiments described in §4 required 2 hours for both PolyLM and LSDP models, 34 hours for direct prompting of Llama models, and 7.5 hours for GPT-4o. The experiments in §5 took 4 hours, while those in §6 required 20 hours.

Model	Size on GPU	τ
Llama 3.1 8B	16GB	40m
Llama 3.3 70B	135GB	-
Llama 3.3 70B 4bit	38GB	3h40m
BERT-large[17] + AG_s	3GB	5m45s
GPT-4o	UNK	45 min

Table 10: GPU usage of each model, where τ represents the SemCor-WSI processing time. **Size on GPU** indicates the model's size, which may double during inference. All values are approximate and may vary slightly.

D Modification of the Amrami and Goldberg (2019) algorithm

We modified the process of determining strong and weak senses of the LSDP algorithm. Specifically, strong senses are defined as senses that dominate at least 2 instances (with 2 a hyperparameter tuned by the authors). When a lemma has less instances than the default number of senses, a scenario not occurring for the data tested by the authors, it is possible that each sense would be dominant exactly once (or less). We tested two strategies to manage this scenario: 1) consider all senses as weak, clustering all instances together, and 2) consider all senses as strong, placing them in separate clusters. The former strategy yielded better results, and thus, we report only them in the table 3.

E SemCor-WSI dataset statistics

In Table 11, the statistics of the SemCor-WSI dataset are provided in comparison with the original SemCor corpus.

POS	Dev set			Test set			SemCor
	Inst.	Lem.	Polysemy	Inst.	Lem.	Polysemy	Polysemy
Adj	4909	433	1.69[±1.3]	4772	427	1.69[±1.2]	1.64[±1.2]
Noun	5394	479	1.75[±1.4]	5694	493	1.73[±1.4]	1.71[±1.4]
Verb	5005	359	2.39[±2.2]	4979	367	2.36[±2.4]	2.34[±2.5]
All	15308	1271	1.94[±1.7]	15445	1287	1.91[±1.7]	2.1[±2.2]

Table 11: SemCor-WSI dataset statistics (no hapaxes).

F Proportion of each POS in SemCor 3.0

In table 12, we report the percentage of each part of speech in SemCor 3.0 dataset. These values were used to compute the $_w$AVG metric reported in Table 3.

POS	%
Noun	0.49
Adjective	0.22
Verb	0.30

Table 12: Percentage of each POS in SemCor Brown1 and Brown2 from which SemCor-WSI was composed.

[17] Number of parameters of BERT-large-uncased: 334M

G Best layer for each model in Tables 3, 4 and 5

In Tables 13, 14 and 15, we provide the results of tuning the *layer* hyperparameter for each model tested.

Model	ALL	Verb	Adj	Noun
AG$_{silh}$				
BERT-l-Wikt	23	23	24	23
BERT-b-u	11	10	11	11
MirrorWiC-base	9	9	9	9
BERT-l-u	20	17	19	22
X-Means				
BERT-l-Wikt	21	21	23	23
BERT-b-u	12	12	12	12
MirrorWiC-base	12	11	12	11
BERT-l-u	21	24	3	24

Table 13: for Table 3, best layer for each PLM on AG$_{silh}$and X-means.

Aug	**Base**	**Must-link**	
	AG$_s$	**AG$_{wikt}$**	**AG$_s$**
No	20	NA	NA
Wiktionary	20	17	20
Llama 3.1 8B 4bit	24	22	17
GPT-4o	24	19	24
WB (10 per l.)	16	24	22
WB (50 per l.)	16	23	16
WB (100 per l.)	19	20	19
WB (150 per l.)	20	20	18

Table 14: For Table 4, best layer for each data augmentation type on BERT-l-u + AG.

Aug	**Base**	**Must-link**	
	AG$_s$	**AG$_{wikt}$**	**AG$_s$**
No	23	NA	NA
Wiktionary	23	23	23
Llama 3.1 8B 4bit	24	23	22
GPT-4o	24	23	23
WB (10 per l.)	23	22	20
WB (50 per l.)	24	22	20
WB (100 per l.)	22	22	22
WB (150 per l.)	20	22	20

Table 15: For Table 5, best layer for each data augmentation type on BERT-l-Wikt + AG.

H Statistical significance: bootstrapping

To evaluate the statistical significance of F-B^3 results differences, we apply the bootstrapping test (Dror et al., 2020). Being computationally intensive, we only computed statistical significance for all pairs of systems of Table 3 in combination with AG$_{silh}$ as X-means would require rerunning each run 5 times due to its non-deterministic nature. For each pair of models, we verify the null hypothesis that the results difference between two models is due to chance. We sample with replacement the development set of SemCor-WSI 1000 times and perform clustering on each sample using both models. Then, for each sample, we compute the difference between two models' performance (Δ_{sample}) and check if it is higher than twice the original difference between the 2 models (Δ_{obs}). The p-value is the proportion of samples for which $\Delta_{sample} \leq 2 \times \Delta_{obs}$. We reject the null hypothesis when p-value is less than 0.05. In Figure 1, we present the histograms of bootstrap results for each PLM in combination with AG$_{silh}$.

I Statistics of added examples from each source

The total number of examples generated or gathered from WikiBooks is presented in Table 16. For WikiBooks, we note that the number of examples is not equal to the number of lemmas × the number of additional examples, as some lemmas are missing from the corpus and some had less examples than required. For LLMs, the number is not equal to the number of instances of SemCor-WSI × the number of generated examples, as both models occasionally generated more or less than 3 examples, or refused to perform the task at all.

Source	Selection	Verb	Noun	Adj	Total
Wikt	all per L	2705	4016	1805	8526
Llama	3 per Inst.	15171	16439	14852	46462
GPT-4o	3 per Inst.	14970	16154	14700	45824
WB	10 per L	3511	4184	3539	11234
WB	50 per L	16910	19341	15246	51497
WB	100 per L	32457	36421	27606	96484
WB	150 per L	46671	51994	38756	137421

Table 16: Total number of examples added for each POS using different sources. **per l** = N examples added for each lemma, **per inst** = N examples added for each instance.

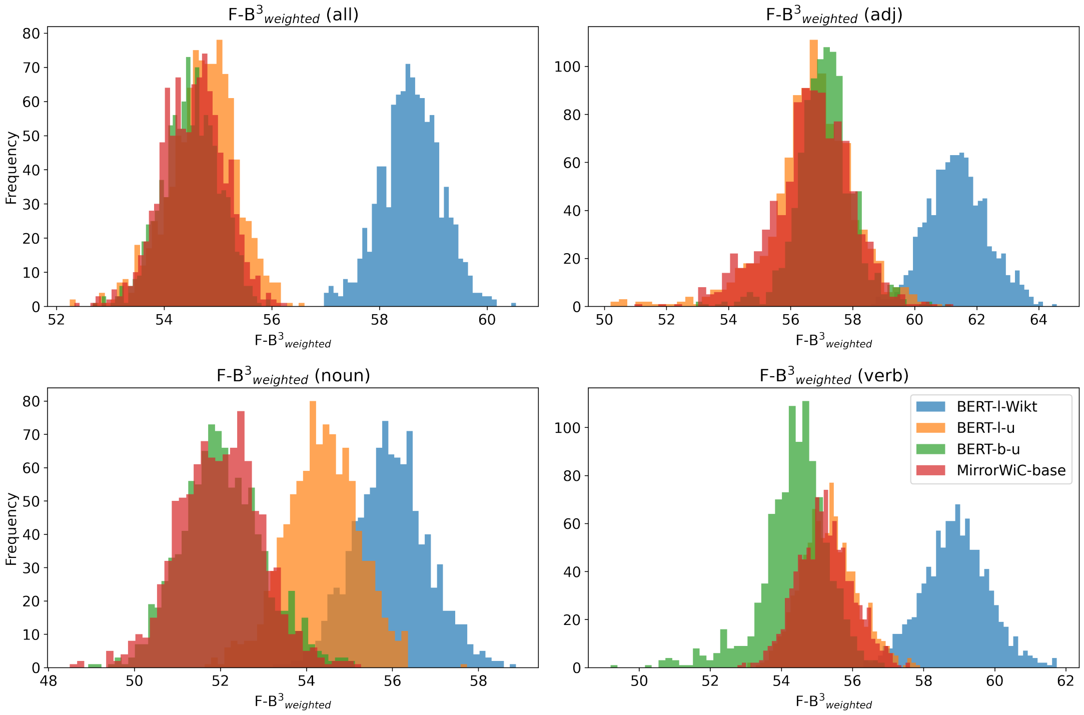


Figure 1: Bootstrapping distribution for 1000 runs using AG$_{silh}$ in combination with all PLMs.

J LLMs details and prompts for direct WSI prompting

We tested 3 large language models: the proprietary GPT 4-o (gpt-4o-2024-08-06) (OpenAI et al., 2024) and two open-source models: Llama 3.1 8B Instruct[18](Grattafiori et al., 2024) and Llama 3.3 70B Instruct (4 bit)[19].

For the direct WSI prompting, we tested three prompt strategies, where the LLM was asked either to: 1) provide a Python list with cluster numbers for each instance, 2) arrange instance identifiers into Python lists considered as clusters, 3) or assign a sense identifier for each sentence with its index. The last approach yielded the best results, thus, we provide the corresponding prompts below and the results for this approach only.

For SE13, we tuned the prompt using its trial set. For SE10 and SemCor-WSI, the prompt was tuned using the SemCor-WSI development set, as the task for both datasets is to predict a single sense per instance. We set the maximum sequence length to 40,000 (to handle lemmas with 500+ instances), the maximum number of new tokens to 4,000, and the default values for the remaining hyperparameters. Model responses were parsed using regular expressions, and missing values were assigned a uniform dummy sense identifier.

J.1 SemEval 2010 Task 14 and SemCor prompt:

```
Given the following examples of sentences using the lemma '[LEMMA]', identify the sense of the target lemma for each sentence.
  Examples:
  —
  [[INDEX]. [SENTENCE]]
  —
  For each sentence, your response should be in the format: '[sentence_index]. [sense_identifer]'.
  Please respond with one sense for each sentence. Please provide answers for all examples. Do not write any explanations. Do not write the sentence in your answer. Only give the sentence index and sense identifier.
```

J.2 SemEval 2013 Task 11 prompt:

```
Given the following examples of sentences using the lemma '[LEMMA]', identify the possible senses of the lemma and their level of applicability for each
```

[18] Version released on July 23, 2024, https://huggingface.co/unsloth/Meta-Llama-3.1-8B-Instruct

[19] Version released on Dec 6, 2024, https://huggingface.co/unsloth/Llama-3.3-70B-Instruct. For all Llama models, we use unsloth (Daniel Han and team, 2023) for faster computation.

sentence. For each sentence, list the possible senses of the lemma with their corresponding level of applicability (from 0 to 1).

Examples:

——

[[INDEX]. [SENTENCE]]

——

For each sentence, your response should be in the format: '[sentence_index]. [sense_1/applicability_1] [sense_2/applicability_2']'. For example, the answer might look like "100. sense_1/2", "100. sense_1/0.8 sense_2/0.4" or "100. sense_1/1 sense_2/0.4 sense_3/4".

Please respond with the possible senses of the lemma and their level of applicability for each sentence. Do not write any explanations. Do not write the sentence in your answer. Only give the sentence index, sense identifiers and their level of applicability.

K Prompt for generating unlabeled new examples

Create 3 examples with the target lemma '[LEMMA]' where this lemma is used in the same sense as in the sentence '[SENTENCE]'. Separate each example by \n and do not give any explanations.

L Hyperparameters for clustering experiments of §5.2

We used the scikit-learn implementation of Agglomerative Clustering and silhouette score (Pedregosa et al., 2011), with following hyperparameters: average linkage with euclidean distance, minimum number of clusters for silhouette score is 2, maximum is 15. More precisely, for a lemma having n instances, silhouette is only defined for numbers of clusters c such as $2 \leq c \leq n-1$. So if $n \geq 3$, we select the number of clusters $c^* = min(15, \text{argmax}_{2 \leq c \leq n-1} silh(c)$. If $n = 2$, we return a single cluster.

For X-means, we used the pyclustering implementation (Novikov, 2019), with following hyperparameters: minimum number of clusters is 1, maximum is 15, tolerance is 0.003.

M Table 2: results for each POS

In Table 17, we detail the performance of each model from Table 2 for each part of speech. We note that the results for adjectives from SE10 are absent, as they were not included in the test set.

N Datasets and LLMs Licenses

In our experiments, we use WikiBooks part of the BigScience corpus distributed under the BigScience RAIL License available at: https://huggingface.co/spaces/bigscience/license. Additionally, we use the Wiktionary DBnary dataset, released under the Creative Commons Attribution-ShareAlike 3.0 license. We introduce a new evaluation framework (SemCor-WSI) based on SemCor 3.0, which is the property of Princeton University. The corresponding license is included within the SemCor package, accessible at http://web.eecs.umich.edu/~mihalcea/downloads/semcor/semcor3.0.tar.gz.

Considering Llama models, we use Llama 3.1 (license available at https://www.llama.com/llama3_1/license/) and Llama 3.3 models (license available at https://www.llama.com/llama3_3/license/).

Model	**SemEval 2013**		**SemEval 2010**				**SemCor-WSI**	
	Fuzzy-NMI	**Fuzzy-F-B^3**	**V-M**	**Paired F-S**	**NMI**	**F-B^3**	**F-B^3**	**NMI**
				Verb				
PolyLM large	**25.6**	**67.8**	45.2	**75.6**	4.5	58.7	68.5	28.9
PolyLM base	25.2	66.5	**45.3**	71.9	4.3	54.2	65.8	27
LSDP	18.5[±0.6]	59.1[±0.8]	43.2[±1.1]	66.0[±0.8]	4.2[±0.2]	60.2[±0.3]	65.2[±0.3]	23.6[±0.7]
Llama 8B	2.3[±0.5]	57.7[±0.5]	13.2[±0.7]	53.1[±1.9]	6.0[±0.3]	54.5[±1.3]	57.8[±1.8]	19.1[±1.0]
Llama 70B	7.1[±0.6]	41.2[±2.3]	23.3[±0.7]	59.5[±3.1]	5.6[±0.3]	57.2[±2.6]	**68.7[±0.3]**	**34.2[±0.6]**
GPT-4o	15.0[±1.1]	57.3[±2.3]	33.8[±2.7]	67.6[±4.9]	5.0[±0.4]	51.0[±4.8]	63.1[±1.7]	26.1[±0.7]
1cpl	0	61.5	0	72.7	0	**73.4**	65.7	14
1cpex	7.1	0	25.6	0.1	**15.7**	8.2	25.5	26.9
				Noun				
Model	**SemEval 2013**		**SemEval 2010**				**SemCor-WSI**	
	Fuzzy-NMI	**Fuzzy-F-B^3**	**V-M**	**Paired F-S**	**NMI**	**F-B^3**	**F-B^3**	**NMI**
PolyLM large	**23.4**	**64.5**	42.5	62	7.3	42.7	74.3	**41.9**
PolyLM base	20.5	62.5	39.3	62.7	7.5	45.6	72.9	38.2
LSDP	22.2[±0.6]	64.3[±0.5]	**47.1**[±1.1]	**67.4**[±0.6]	4.8[±0.2]	47.8[±0.4]	72.3[±0.7]	36.5[±1.1]
Llama 8B	2.2[±0.5]	58.1[±1.5]	18.7[±1.3]	46.6[±1.7]	8.1[±0.7]	46.3[±2.0]	60.0[±1.0]	22.7[±0.9]
Llama 70B	10.8[±0.4]	46.2[±2.1]	33.6[±1.2]	42.9[±6.2]	9.8[±0.8]	44.3[±1.1]	65.5[±2.4]	23.8[±3.9]
GPT-4o	18.3[±1.2]	59.9[±1.3]	38.1[±2.0]	61.3[±1.5]	8.5[±0.4]	45.4[±0.7]	71.4[±0.6]	37.3[±1.1]
1cpl	0	61.8	0	57	0	**57.6**	**75.2**	30.4
1cpex	7.1	0	35.8	0.1	**22.1**	7.9	24.1	19.1
				Adjective				
Model	**SemEval 2013**		**SemEval 2010**				**SemCor-WSI**	
	Fuzzy-NMI	**Fuzzy-F-B^3**	**V-M**	**Paired F-S**	**NMI**	**F-B^3**	**F-B^3**	**NMI**
PolyLM large	20.7	68	NA	NA	NA	NA	76	29.1
PolyLM base	23.8	**68.7**	NA	NA	NA	NA	74.9	27.1
LSDP	**24.4**[±1.3]	62.4[±0.7]	NA	NA	NA	NA	75.5[±0.4]	35.8[±1.1]
Llama 8B	2.6[±0.6]	53.6[±1.7]	NA	NA	NA	NA	61.4[±1.8]	15.9[±1.7]
Llama 70B	8.6[±0.7]	45.5[±2.8]	NA	NA	NA	NA	58.0[±2.2]	24.9[±0.5]
GPT-4o	18.0[±1.8]	58.1[±4.2]	NA	NA	NA	NA	65.7[±0.9]	23.5[±5.3]
1cpl	0	59.4	NA	NA	NA	NA	**80**	**39.8**
1cpex	6.6	0	NA	NA	NA	NA	22.6	16.2

Table 17: Extension of Table 2 for each part of speech subset.