Research Article

Vítor Míguez-Rego*

Large language models as first-pass filters for corpus annotation: semantic disambiguation of Galician *pobo*

https://doi.org/10.1515/opli-2025-0078
Received September 22, 2025; accepted January 8, 2026; published online February 9, 2026

Abstract: This paper demonstrates the use of LLMs as first-pass filters in corpus annotation, with a focus on semantic disambiguation – a task more challenging than form-based classification due to its context-dependence. Using as a case study the polysemous Galician noun *pobo* 'people/village', the study demonstrates the applicability of LLM-assisted annotation to low-resource languages. 300 examples were annotated by three human coders and four LLMs (Claude 4 Sonnet, Claude 4 Opus, Claude 4.5 Sonnet, and Claude 4.5 Opus) using a static, single-phase prompting approach. Since first-pass filters should capture as many actual occurrences of the target phenomenon as possible, priority was given to recall over precision. Accordingly, the paper argues for F_2, a recall-focused metric, over commonly used alternatives like F_1 or MCC for validating LLM performance in filtering tasks. Claude 4.5 Opus with pretraining achieved the best performance against the human consensus (F_2 = 0.944, recall = 100 %), resulting in substantial workload reduction with no information loss. The study demonstrates that LLMs can serve as effective first-pass filters for semantic annotation in corpus linguistics, extending their applicability to low-resource languages.

Keywords: corpus linguistics; corpus annotation; semantic disambiguation; large language models; Galician

1 Introduction

In the last decades, linguistics has experienced a series of transformations that have been understood as a quantitative turn (Kortmann 2021). The number of studies using statistical tools to explore linguistic data and test hypotheses has increased, as has the sophistication of those statistical methods. Thus, multifactorial predictive techniques, such as regression and tree-based models, have become *de facto* standards in many fields. Corpus-based variationist studies are a good example of this shift (Gries 2025).

Parallel to this development in the field of statistics, the automatic annotation of the texts that make up a corpus has become more accurate in several areas, including part-of-speech tagging, lemmatization, parsing, and semantic annotation (Newman and Cox 2020, p. 35). However, these advancements have not spread equally across corpora and languages, and even the most complete corpora in major languages fail to capture many aspects of interest to linguists. Therefore, after extracting data from a corpus, researchers often need to filter out false positives (all those cases that were retrieved in the corpus query but do not correspond to their object of study) or to manually annotate a number of linguistic features relevant for their investigation. This is particularly challenging for semantic annotation, where context-dependence makes automation difficult, and for low-resource languages, where natural language processing (NLP) tools are less developed. Nevertheless, the recent breakthroughs in artificial intelligence and, particularly, the development of powerful large language models (LLMs) offer an opportunity to speed up annotation tasks. If successful, such an approach would not only drastically

***Corresponding author: Vítor Míguez-Rego**, Universidade de Santiago de Compostela, Instituto da Lingua Galega, Praza da Universidade, 4, 15782, Santiago de Compostela, Spain, E-mail: vitor.miguez@usc.gal. https://orcid.org/0000-0001-7138-373X

decrease the amount of time needed to carry out a corpus-based investigation, but it would also increase the reliability of its results by streamlining the annotation process and reducing human errors and inconsistencies while maintaining human oversight.

In order to explore the potential of LLMs for corpus annotation in an ecologically valid context, this paper will start from a case study of the semantics of the Galician noun *pobo* 'people'. Mariño Paz (2024) has recently studied this lemma from a diachronic perspective with a focus on the emergence and spread of a semantic innovation consisting in using *pobo* to describe a human settlement, usually a town or a village. This use was introduced into Galician in the medieval period via translations from Spanish and became widespread in the 19th century. In present-day Galician, this use of *pobo* is considered a semantic calque from Spanish and banned from the standard variety. However, it can still be found in samples of present-day language and, as Mariño Paz (2024) points out, it appears to prevail in contexts where linguistic revision of texts has little to no role to play, such as press and speech.

To test the hypothesis that the use of *pobo* as 'settlement' is more prevalent in more spontaneous text types, one could rely on a corpus linguistic (CL) approach to investigate the distribution of *pobo* and its standard equivalents *vila* 'town' and *aldea* 'village' across different genres. Crucially, after retrieving the relevant occurrences of *pobo* from a corpus, one must filter out all those cases where *pobo* does not describe a place, but has its standard meaning of 'people'. This preprocessing of the data would likely be the most time-consuming phase in such an investigation. In an effort to reduce the time required to complete this step, this paper will focus on the use of LLMs as first-pass filters in the annotation of corpus data.

Pioneering research on NLP shows that LLMs can outperform human annotators: Gilardi et al. (2023) found that ChatGPT's accuracy exceeds human annotators by around 25 % points and is 30 times cheaper. LLMs are also effective in word sense disambiguation tasks (Yae et al. 2025), which is particularly relevant for semantic annotation challenges like distinguishing between different meanings of a polysemous word like *pobo*. There are also recent contributions in the CL literature that show the advantages of relying on LLMs to (pre)annotate data. Yu et al. (2024) explored the annotation of complex pragmatic and discursive features through LLMs, whose performance fell slightly short of that of a human coder. Morin and Marttinen Larsson (2025a) achieved over 90 % accuracy in the detection of particular grammatical constructions in large corpora through an iterative process. In a subsequent study (Morin and Marttinen Larsson 2025b), they redesigned their approach as an unsupervised pipeline, scaling it to over 140,000 sentences using API-based batch processing.

This paper addresses three gaps in the emerging literature on LLM-assisted corpus annotation. First, existing studies have focused predominantly on English and other high-resource languages, leaving open the question of whether LLMs can effectively process low-resource languages like Galician. Second, previous work has prioritized form-based annotation tasks, such as syntactic classification, over semantic disambiguation, which is inherently more challenging due to its context-dependence. Third, model validation has relied on general-purpose metrics like F_1 and MCC that weigh precision and recall equally, despite the fact that filtering tasks in corpus linguistics should prioritize recall to avoid information loss. By exploring these issues through a case study on the polysemous Galician noun *pobo*, this paper contributes to establishing methodological guidelines for LLM-assisted annotation in corpus linguistics. All materials used in this study, including prompts and annotated datasets, are openly available in an online repository (see Data availability statement).

2 Corpus linguistics and large language models

LLMs are a very recent development, but they have attracted considerable attention across diverse fields of knowledge due to their applicability to a wide range of scenarios. In the field of CL, the focus has been on how to use LLMs as copilots for linguists (Torrent et al. 2024). In this connection, Yu et al. (2024) explored the use of LLMs to automate the annotation of pragmatic and discursive aspects in corpus data. This kind of annotation is particularly challenging, since many pragma-discursive features lack direct mapping to lexical forms. The researchers compared the performance of a human coder and two LLMs (GPT-3.5 and GPT-4 from OpenAI) on apology speech acts in English. Using a local grammar framework, they annotated more than 5,000 sentences

containing the word *sorry* from a spoken corpus. The annotation scheme included five functional elements: *apologizing* (the apology expression), *reason* (why someone is apologizing), *apologizer* (who apologizes), *apologizee* (the recipient of the apology), and *intensifier* (expressions that boost the degree of the apology). In prompt design, they used a few-shot prompting technique, thus including examples of the categories to enrich task instructions. They found that GPT-4 clearly outperformed GPT-3.5 on a sample of 50 instances, with an accuracy of 84 % against 50 % at the instance level (an instance was considered accurately annotated when all tags were correctly coded). Then, they compared the performance of GPT-4 to that of a human coder on a dataset of 1,000 examples, with one of the authors serving as an assessor. At the instance level, the accuracy of the LLM fell slightly short of that of the human coder (92.7 % vs. 95.4 %). At the tag level, performance varied systematically depending on the linguistic characteristics of the functional elements. Notably, GPT-4 slightly outperformed the human coder in the annotation of *reason* (F_1 = 0.912 vs. 0.893), and demonstrated strong comprehension and annotation capabilities in general, with the exception of the category of *no apology*, where it clearly underperformed. The authors consider that these results show that LLMs are a viable option to assist in the task of annotating apologies and other speech acts, but human oversight remains necessary.

In another recent paper, Morin and Marttinen Larsson (2025a) used the LLM Claude Sonnet 3.5 (Anthropic) to identify the relevant cases of the English evaluative construction "*consider* X (as) (to be) Y" among more than 18 million tokens of *consider* retrieved from a large corpus. The authors developed a replicable three-step process based on prompt engineering, iterative training, and evaluation. Their methodological pipeline starts with formulating input prompts that provide clear instructions and examples, include XML tags to help with prompt processing, and ask the model to think about every decision (known as "chain-of-thought" reasoning, see Wei et al. 2022) in order to improve the model's performance. Then, iterative training follows, which includes pretraining (the model looks at a number of annotated sentences to analyze classification patterns) and supervised training with corrective feedback (the model analyzes small batches of sentences and receives corrections). Finally, the model is tested on unseen data, taking into account the insights gained from the previous rounds of iterative training. This process allowed the researchers to reach an accuracy of 93 % on a dataset of 101 sentences, with a strong performance according to other evaluation metrics, including precision, recall, F_1 and Matthews correlation coefficient (MCC).

These contributions represent two different applications of LLMs to CL and illustrate contrasting methodological philosophies. Yu et al. (2024) used a static few-shot approach with fixed examples to test the LLM's capabilities with minimal intervention. Morin and Marttinen Larsson (2025a) also included fixed examples, but relied on dynamic iterative training with corrective feedback to actively shape LLM performance. Yu et al. (2024) used a single-phase prompt design with trial-and-error refinement, whereas Morin and Marttinen Larsson (2025a) relied on a multi-stage pipeline. Despite these substantial methodological differences and the varying requirements of the tasks they addressed, the performance of LLMs was strikingly similar, approaching an accuracy of 93 % in both cases, which demonstrates the remarkable versatility of this technology.

Both studies converge on the critical importance of prompt engineering, providing key recommendations to improve performance. On the one hand, Yu et al. (2024) identified eight factors, besides including examples, that affect the LLM's performance: the formal layout of the prompt, its terminological precision, conciseness and explicitness, the grammatical correctness of the examples, the clarity of labels, and the presence of inappropriate language. On the other hand, Morin and Marttinen Larsson (2025a) also emphasize the importance of including examples and formulating a clear, specific and contextualized prompt, while advocating for XML structuring and chain-of-thought reasoning. As for the processing of examples, Yu et al. (2024) processed each instance individually, while Morin and Marttinen Larsson (2025a) instructed the model to work in small batches of sentences (20–25). These considerations suggest that successful LLM-based annotation requires some methodological sophistication, despite the technology's accessibility and apparent ease of use.

These contributions differ in a crucial aspect of prompt design. Yu et al. (2024) chose a static, single-phase approach. Thus, once they developed their prompt, they used it consistently throughout the annotation process. In turn, Morin and Marttinen Larsson (2025a) opted for a dynamic, iterative approach where the model receives ongoing feedback during the annotation process. Each of these approaches presents advantages and limitations. The single-phase approach favors consistency by employing the same prompt across all instances, thus

eliminating the variability that could come from changing instructions. It is scalable: once developed, requires no additional human intervention during annotation. It favors reproducibility, as the original conditions are easy to replicate by other researchers. But these advantages come at a cost. Single-phase prompting gives up learning during annotation, since it cannot take into account patterns discovered during the process, thereby requiring extensive upfront investment in prompt engineering. Success depends entirely on the quality of the prompt, and in large datasets edge cases are more likely to be missed during the prompt design stage.

In contrast, the iterative approach to prompt design allows for adaptation and learning from errors during annotation. This progressively improves performance of the model as it encounters more examples. There is also a lower risk of missing edge cases in large datasets, because they are more likely to be found and do not have to be anticipated. However, the iterative approach requires continuous human intervention and may be more time-intensive due to the feedback loop process. It may also generate inconsistencies as the model's understanding of the task evolves during annotation. For instance, there may be a risk of overfitting to early corrections.

Nevertheless, the most serious limitation of the iterative approach is the fact that it does not take into account that LLMs have finite context windows. Iterative training with corrective feedback likely accumulates significant conversation history, thus causing the model to reach its conversation limit sooner. Once the limit is exceeded, the researcher must begin a new conversation and restart the iterative training process. In this regard, Morin and Marttinen Larsson (2025a) do not report how many examples could be analyzed in a single conversation before reaching the model's memory limit. It took them approximately 60 min to complete the process from pretraining to the positive evaluation of 102 examples, but in a real annotation setting with thousands or even millions of instances to process this would predictably involve multiple interrupted sessions. In their follow-up study, Morin and Marttinen Larsson (2025b) addressed the scalability problem by adopting a static, unsupervised approach. Their four-phase pipeline achieves more than 98 % accuracy on over 140,000 sentences without iterative training or corrective feedback, demonstrating that comprehensive upfront prompt engineering can substitute for ongoing human supervision.

Beyond the choice of prompting strategy, implementation method is another practical consideration. To process data at scale, Morin and Marttinen Larsson (2025b) accessed the LLM via API rather than a conversational interface, enabling automated batch processing of tens of thousands of sentences with minimal programming. Web-based interfaces require no programming at all and offer a lower barrier to entry, though they involve manual submission of batches. These two implementation strategies represent complementary options: API access favors scalability and automation, while web-based interfaces favor accessibility and ease of use. The present study adopts a web-based approach, prioritizing accessibility for linguists without programming experience.

Another important issue in this emerging field concerns the role of LLMs in the CL workflow. Both Yu et al. (2024) and Morin and Marttinen Larsson (2025a) argue for a hybrid approach where LLMs perform particular tasks under human supervision. Morin and Marttinen Larsson (2025b) refine this model by eliminating supervision during annotation while maintaining human oversight through pre-hoc and post-hoc validation. Yu et al. (2024, p. 553) suggest using LLMs "as a 'first pass' technique to automatically generate tentative annotations to be validated by a human coder". Morin and Marttinen Larsson (2025a, 2025b) also use an LLM as a first pass, particularly as an initial filter designed to identify which data points are true cases of the construction under investigation, with researchers subsequently validating these machine-generated classifications before proceeding with their analysis. In these hybrid workflows, humans play a crucial role as the ultimate decision-makers, and retain responsibility over the final result, guaranteeing scientific accountability and methodological rigor.

A key aspect of CL workflows relying on human-LLM collaboration is the choice of metrics to evaluate the machine's performance. Yu et al. (2024) and Morin and Marttinen Larsson (2025a, 2025b), following standard practice in NLP, rely on accuracy, precision, recall, F_1 and/or MCC. While computing and reporting all these metrics is very informative, the final validation of a model usually relies on F_1 or MCC, as they combine precision and recall into a single value with an intuitive interpretation (for a comparison of these and other metrics, see Chicco et al. 2021). However, these "general-purpose" metrics may not capture the practical implications of the human-machine collaboration for all CL workflows, since they use an even weighting for precision and recall. In

other words, the costs of different error types might substantially differ depending on the CL task at hand. For example, an equal weighting for precision and recall, which gives the same importance to false positives (cases incorrectly assigned to a category) and false negatives (cases incorrectly missed from a category), may be reasonable when the machine is used to provide tentative annotations to be validated manually, as in Yu et al.'s (2024) study. Nevertheless, when the cost of a false negative clearly outweighs that of a false positive, recall should be prioritized so as to avoid missing crucial information.

The emphasis on recall over precision is a constant across a number of fields. The classic example is medical diagnosis, where missing a positive case could have dire consequences compared to misidentifying a negative case as positive. For instance, Burkow et al. (2024) present deep-learning methods to improve the detection of pediatric rib fractures in chest X-rays. Since producing false negatives (i.e., missing rib fractures) has severe implications, the researchers prioritized recall, which measures coverage of the phenomenon, and relied accordingly on F_2, a metric that weighs recall more heavily than precision. The use of F_2 as an evaluation metric is widespread in other areas where detection of all potential cases of a phenomenon is paramount: ranging from construction accident investigation (Uhm et al. 2025) to credit card fraud identification (Zhao et al. 2024), and from ransomware monitoring (Ispahany et al. 2025) to anomaly recognition in additive manufacturing (Mattera et al. 2024), many recent works use F_2 as their primary evaluation metric.

This suggests that CL workflows using LLMs for filtering or detection tasks may similarly benefit from a recall-focused evaluation metric like F_2. Choosing F_2 over F_1 or MCC implies sacrificing precision in favor of recall. In the context of a CL filter, this means that the researcher is willing to risk spending additional time reviewing irrelevant cases to ensure they capture as many actual occurrences of the target linguistic phenomenon as possible. This trade-off aligns with CL priorities, where missing relevant instances can compromise the validity of the generalizations drawn from the corpus (Stefanowitsch 2020, Chapter 2). In contrast, reviewing additional examples that end up being discarded, while time-consuming, does not threaten the integrity of the final dataset. Most corpus linguists would likely see this extra annotation effort as a worthwhile investment to ensure comprehensive coverage of a linguistic phenomenon, especially considering that the use of an automatic filter already represents a (possibly substantial) improvement over fully manual annotation.

Given the above considerations about prompt engineering, the role of LLMs in the CL workflow, and evaluation metrics, a final question arises about the generalizability of LLM-based annotation. Both Yu et al. (2024) and Morin and Marttinen Larsson (2025a, 2025b) identified several key areas in this regard. In terms of empirical scope, these studies worked on a single linguistic phenomenon (*sorry*-based apologies and evaluative *consider* constructions, respectively) and explicitly acknowledged that research on other linguistic phenomena is necessary to demonstrate broader applicability. This is critically related to a core issue regarding the linguistic complexity of the object of study. The works under review found that LLM performance is inversely proportional to the complexity of the target phenomenon, since formulaic, form-based patterns achieved higher scores than context-dependent, semantic features. Yu et al. (2024) and Morin and Marttinen Larsson (2025a) emphasize that annotation tasks using LLMs require substantial adaptation for different phenomena, limiting direct transferability across studies. Morin and Marttinen Larsson (2025a) caution that LLMs "should only come into play after substantial preparatory work on the grammatical construction, including a comprehensive understanding of the envelope of variation and its possible edge cases."

A final generalizability concern regards the cross-linguistic applicability of LLM-based annotation workflows. LLMs are trained on massive amounts of text data scraped from a variety of digital sources, where English and, to a lesser extent, other high-resource languages, such as Chinese, German, Russian, Spanish, or French, are vastly overrepresented. Low-resource languages constitute a minuscule fraction of LLM training datasets, which in some contexts leads to poor performance compared to languages with vast resources (Tessema et al. 2024). However, despite this imbalance, state-of-the-art LLMs use extensive pretraining across many languages, enabling them to create coherent outputs even when training data is scarce and giving researchers "access to more powerful and versatile tools for processing and analyzing low-resource languages" (Zhong et al. 2024). CL is a method that can be used to investigate a plethora of phenomena from any given language. Since LLM applications to CL are still in their infancy, future empirical work will determine the true scope and limitations of LLM-assisted CL workflows.

Based on the methodological insights and gaps identified in the literature, this study addresses the following research questions:

(1) How reliable are LLM annotations compared to manually-produced ones?
(2) What are adequate metrics for validating LLM-based annotations in CL filtering tasks?
(3) Do larger, more computationally expensive LLMs perform better than smaller, more efficient ones in semantic annotation tasks?
(4) Does pretraining improve LLM annotation performance?
(5) What time efficiency gains can be obtained by using LLMs as first-pass filters in CL workflows compared to fully manual annotation?

By exploring these research questions through a case study on semantic disambiguation in Galician, this paper seeks to contribute to the emerging field of LLM-assisted CL.

3 Methods

3.1 Corpus and data extraction

The data for this study was extracted from the *Corpus de Referencia do Galego Actual* (CORGA) Version 4.1, the main reference corpus for present-day Galician, freely available on the web. CORGA 4.1 includes more than 45 million words from 1975 to 2024, and is balanced between fiction (dramatic and narrative prose), newspapers, and essays, with a small oral component.

The corpus was searched for the orthographic string <pobo|pobos> with no case sensitivity, thus retrieving the orthographic forms *pobo*, *pobos*, *Pobo*, and *Pobos*. A chronological filter was applied, restricting results to the period from 1995 to 2019, in order to ensure adequate dispersion across documents. CORGA's composition varies substantially over time: the period 1975–1994 averages approximately 10,000 words per document, whereas 1995–2019 averages around 700 words per document. This means earlier periods are dominated by a small number of long texts, which would compromise the representativeness of any random sample. The selected period (1995–2019) provides both sufficient data volume and adequate document dispersion. A total of 6,293 examples were obtained and downloaded in CSV format, with each row containing one example and its metadata.

In CSV files, examples are divided in three columns, with one column containing the target expression (*pobo* or one of its variants) and the previous and next columns the left and right context, respectively. Preserving this structure is important, since it clearly informs the annotator of what the target expression is, and it becomes vital when several instances of the target expression are present in the same concordance. Although CORGA is lemmatized, search by lemmas was avoided because the downloaded examples of the lemmatized search mode are syntactically parsed. In Galician, where prepositional and clitic contractions are frequent, parsed examples may interfere with accurate linguistic processing.

Finally, 400 examples were randomly selected from the dataset to be annotated: 300 to serve as the testing dataset for both human annotators and LLMs, and an additional 100 for the pretraining dataset used in one experimental condition. The sample was not stratified by time period, as the present study focuses on synchronic variation across genres rather than diachronic change. While previous research has traced the historical development of the 'settlement' sense of *pobo* (Mariño Paz 2024), the goal here is to test LLM performance on a semantic disambiguation task in present-day Galician.

3.2 Annotation task design

As established in the introduction, *pobo* exhibits polysemy between several standard meanings, roughly equivalent to *people* in English, and an innovative 'settlement' meaning resulting from contact with Spanish. The context of this disambiguation task is a corpus study of the conditional distribution of the 'settlement' sense of

pobo and its standard variants *vila* and *aldea*. Thus, the task at hand simulates a filtering stage in the CL workflow whose goal is the correct identification of all corpus instances where *pobo* potentially describes a human settlement (town, village, or similar).

The task consists in the binary classification of each example in the testing dataset as either "keep" or "discard." The discard label is intended for all those instances of *pobo* that clearly do not correspond to the settlement sense, as in the following examples, where *pobo* describes an ethnic or national group or the common people.[1]

(1) a. *Cinco anos de traballo dende a base que fixeron protagonista de todo isto o pobo catalán, poñendo os representantes políticos do noso lado, acompañando o proceso, e que farán que o novo …*
'Five years of work from the base that made the Catalan people the protagonist of all this, putting the political representatives on our side, accompanying the process, and that will make that the new …' (CORGA)

b. *Agromou así o ideal democrático moderno, no que o goberno e as leis deberían xurdir do pobo mesmo e estaren controlados por el.*
'Thus sprang up the modern democratic ideal, in which the government and the laws should emerge from the people itself and be controlled by it.' (CORGA)

The keep label should be assigned to all cases where *pobo* refers to a settlement or may potentially refer to one. This includes cases where context is insufficient to appropriately assess the meaning of the example or the settlement sense coexists with another sense of the word. The following examples illustrate the keep classification:

(2) a. Settlement sense
Aínda hai que recoller os billetes, ir a ese pobo e coller o tren.
'It's still necessary to pick up the tickets, go to that town and take the train.' (CORGA)

b. Insufficient context
Agora di referíndose ó mesmo pobo:
'Now he/she says, referring to the same people/town:' (CORGA)

c. Ambiguous context
Dábanlle certa curiosidade, e tiña que causar impresión pensar en todos eses pobos que un nunca vira nin había ver pero que podían ser tan interesantes.
'They gave him/her a certain curiosity, and it had to cause an impression to think about all those peoples/towns that one had never seen nor would see, but that could be so interesting.' (CORGA)

Especially contentious are those cases where *pobo* refers to a particular town but may implicitly mean its inhabitants. This usually takes the form of the construction *o pobo de* 'the town/people of'. Annotators (both humans and LLMs) are instructed to pay special attention to these cases and classify them according to the above criteria, since both meanings are possible with this construction, as the following examples show.

(3) a. Discard classification
O pobo de Santa Comba deunos un cheque en branco con este apoio maioritario e tentarei darlle a Xallas o que el …
'The people of Santa Comba gave us a blank check with this majority support, and I will try to give Xallas what it …' (CORGA)

b. Keep classification
O primeiro lugar no que se embotellou auga foi no pobo de Evián, nos Alpes Franceses.
'The first place where water was bottled was in the town of Evian, in the French Alps.' (CORGA)

1 The examples correspond to the whole concordance downloaded from CORGA. All translations are mine.

This is an inclusive approach to semantic disambiguation that differs from fine-grained semantic annotation. Rather, it reflects the practical needs of an automatic first-pass filter in CL: since it relies on human supervision of machine-generated results, it prioritizes recall over precision (see Section 2). Thus, all keep classifications produced by an LLM are to be manually revised by the researcher to be confirmed as genuine cases of the target meaning. This manual review process relies on examination of the concordance obtained from the corpus, which on some occasions might provide enough context to ascertain the validity of the classification. However, on other occasions, as some of the above examples illustrate, the human supervisor may need to look up a particular instance in the corpus so as to expand the original concordance and obtain sufficient context to make a final decision. The process of searching examples to produce a final decision lies beyond the scope of this paper, and we assume that it happens in a later stage of the workflow. For the purposes of this paper, human supervision of LLM-generated classifications involves either validating a classification or flagging it as needing further examination. This makes the validation task analogous to the annotation task in terms of time and effort requirements, which will enable us to directly address our fifth research question.

3.3 Human annotation protocol

In order to address the first research question regarding the reliability of LLM annotations compared to manual ones, the testing dataset was coded by three human annotators, including the author of the paper. All of them are professional linguists and native Galician speakers. They were provided with the task instructions, which are available in the supplementary materials, and the spreadsheet containing the testing dataset via email. They were asked to rely solely on the context provided in the spreadsheet to annotate the data and to take note of the time it took them to complete the task. They performed the task independently and did not have access to the annotations of the other coders. Once all annotators completed the task, inter-annotator agreement was calculated and a human consensus was produced based on the majority vote for the purpose of serving as the gold standard. The manual annotation of the testing dataset (300 examples) took an average of approximately 1 h and 45 min to complete. Additionally, a pretraining dataset containing 100 examples was manually annotated by the author.

3.4 LLM annotation setup

In order to carry out the automatic annotation of the testing dataset, four models from the Claude family were used for prompt design and testing: Sonnet and Opus in both Version 4 and Version 4.5 (Anthropic 2025a, 2025b). Within each version, Sonnet balances performance and efficiency, while Opus is the more powerful and computationally expensive model (Anthropic 2025c). The comparison of these models will allow us to address our third research question. Claude's LLMs provide the linguist with a user-friendly web interface that requires no programming expertise and exhibited excellent handling of spreadsheet files, which is crucial for careful linguistic annotation (see Section 3.1). LLM annotation was performed on 17 June 2025 (Version 4) and 3 December 2025 (Version 4.5).

LLM annotation was carried out under two different conditions. The first condition followed a basic prompting strategy consisting of an instruction prompt followed by a testing prompt. The second condition included a training prompt after the instruction prompt where the model was shown 100 classified examples from the pretraining dataset before proceeding to annotate the testing data. I will refer to these conditions as the basic and the pretraining condition, respectively. This experimental design will allow us to answer our fourth research question about the impact of pretraining on LLM performance.

A static, single-phase prompting approach was preferred on account of current memory limits in LLM technology (see Section 2). This approach is inherently scalable: once validated, the same prompt can be applied to any number of examples without modification. Following Yu et al. (2024), I systematically refined the initial prompt through several rounds of trial and error, until one of the models surpassed the threshold of 95 % recall against the human consensus. During this process, the most challenging examples were identified and Claude 4

Sonnet was used to analyze their characteristics, enabling the inclusion of specific guidance for cases requiring special attention. The same prompt was later applied without modification to the Version 4.5 models.

Two different prompt sequences were designed, corresponding to the basic and the pretraining conditions. The prompt design draws on the method developed by Morin and Marttinen Larsson (2025a), adapting their approach to the specific requirements of semantic disambiguation in Galician. The first and lengthiest prompt in both sequences is the instructions prompt, which is divided in three main sections. The first section presents the goal of the task and the characteristics of the dataset(s). Here, the LLM is assigned the role of a research assistant in a linguistic research project, following Morin and Marttinen Larsson (2025a), and the structure of CSV files is explained. The prompt explicitly mentions the priority given to covering all possible cases of the target sense and the fact that manual revision of the classifications will be performed afterwards.

A second section in the first prompt contains the sense definitions with on-point examples. A three-sense distinction is made. Senses (a) and (b) correspond to the standard meaning of *pobo* and are based on the definitions provided in the *Dicionario da Real Academia Galega* (González González, n.d.). Sense (c) is the target, non-standard settlement meaning. All senses are followed by two clear examples that illustrate them. (4) gathers each sense definition.

(4) a. A group of people that share a culture, a language, a religion or other social traits.
b. The common people, as opposed to the government or the elites.
c. A human settlement, usually a small town, a country town or a village.

The third and last section in the instructions prompt contains the analysis. Here, the LLM is asked to think step by step about each example following a chain of thought. If sense (a) or (b) are the only possible interpretation, the example must be classified as “discard.” If sense (c) is a possible interpretation, even coexisting with another sense, the example must be classified as “keep.” If the context is insufficient to determine the meaning of *pobo*, the example must be classified as “keep.” In subsequent steps, the LLM is asked to pay special attention to particularly challenging cases, such as those in (3), and to prioritize recall over precision by choosing the keep classification in case of doubt.

In the instructions prompt, the senses and the analysis are structured by means of XML tags, which are also used to identify other key elements, including examples, the target expression within examples, and classification labels. In following prompts, XML is used to introduce datasets. This system provides a clear structure and makes internal references within and across prompts more consistent. Alongside XML, Markdown is used for basic formatting, namely for italics in metalinguistic expressions.

The prompt was written in English rather than Galician for two reasons. First, LLMs are trained predominantly on English data, and prompts in English may yield more reliable instruction-following. Second, English prompts enhance reproducibility by making the method accessible to the international research community. Notably, the LLM successfully processed Galician input data despite receiving English instructions, which itself demonstrates current LLM capabilities with low-resource languages.

In the pretraining condition, the instructions prompt is followed by a training prompt, where the model is asked to examine 100 examples with correct classifications in the pretraining dataset. The model is instructed to use the analysis described in the previous prompt to think about how the data have been classified and consider whether it agrees with the classifications or not.

In the basic prompting sequence there is no training prompt and the instructions prompt is directly followed by a testing prompt. In the pretraining sequence the testing prompt is third in order, following the training prompt. In the testing prompt, the LLM is instructed to classify 300 examples from the testing dataset, working in batches of 50 examples at a time. Initial testing with a greater number of examples resulted in parsing difficulties. The model must think step by step for each classification using the analysis provided in the instructions prompt and include it in its response. Previous rounds of testing showed that asking the model to omit the analysis from its response negatively affected performance.

After the testing prompt, a series of five prompts ask the model to proceed with the next batch of examples, using explicit numbering so as to ensure correct processing of the data.

3.5 Evaluation framework

Inter-annotator agreement between the three human coders was measured using Fleiss' κ (Fleiss 1971), while pairwise comparisons were calculated by means of Cohen's κ (Cohen 1960). Measuring agreement among annotators is vital, as it indicates the reliability of the annotation process (see Artstein 2017).

Our second research question regards the issue of metric selection in model validation. The standard procedure to assess LLM annotation performance is to compare its results to the ground truth or gold standard, which in our case corresponds to the human consensus (see Section 3.3). This comparison can be performed using a variety of metrics, the most popular of which combine precision and recall into a single value (for CL perspectives on precision and recall, see Stefanowitsch 2020, pp. 111–116; Yu et al. 2024, p. 547). In our context, we are interested in precision and recall of the keep class. Precision is the proportion of correct keep predictions (true positives) out of all predictions made by the model (true + false positives); recall is the proportion of true positives out of all instances of keep in the dataset (true positives + false negatives). In summary:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F_1 and F_2 are popular metrics to validate machine-generated results. They are instantiations of the F measure (see Manning et al. 2008, p. 144), which is the weighted harmonic mean of precision and recall. An even weighting of the F measure corresponds to F_1, and is computed as follows:

$$F_1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Other weightings of the F measure may be used to emphasize either precision or recall. F_2 weighs recall more heavily than precision, as reflected in the following formula:

$$F_2 = \frac{5 \cdot \text{precision} \cdot \text{recall}}{4 \cdot \text{precision} + \text{recall}}$$

Another measure that has been used alongside or in place of F_1 is the Matthews correlation coefficient (MCC), which assumes equal importance of positive and negative cases and is more reliable than F_1 for imbalanced datasets (see Chicco et al. 2021). MCC is computed as follows:

$$\text{MCC} = \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}$$

A CL filtering task must prioritize comprehensive coverage of the phenomenon under investigation and must therefore avoid false negatives (see Section 2). In terms of metrics, this translates into giving more weighting to recall over precision. Previous work on the use of LLMs for corpus data annotation have relied primarily on F_1 and MCC for model validation (Morin and Marttinen Larsson 2025a, 2025b; Yu et al. 2024). The present work will explore how the use of a measure that prioritizes recall, such as F_2, can enrich the model validation process in the context of CL filters. Therefore, results will report precision, recall, F_1, F_2, and MCC.

4 Results

4.1 Human annotation reliability

Three-way inter-annotator agreement was measured using Fleiss' κ, while pairwise comparisons relied on Cohen's κ. According to Landis and Koch (1977, p. 165), κ values in the range 0.41–0.60 demonstrate moderate agreement, while values from 0.61 to 0.80 indicate substantial agreement. For our three human annotators a κ

value of 0.656 was obtained, while pairwise comparisons ranged from 0.625 to 0.731. All values are in the substantial agreement range, which suggests consistent interpretation of the task across all annotators.

The human consensus, serving as the gold standard and established through majority voting, shows the following distribution: discard = 263, keep = 37. This yields a ratio of 7.1:1, an imbalance that reflects the distribution of the senses of *pobo* in the corpus, where the non-standard settlement sense is clearly less frequent than the standard senses of the word.

The three annotators agreed unanimously on 262/300 cases or 87.3 % of the sample, whereas for the remaining 12.7 % a majority agreement was reached between two of three annotators. Notably, agreement patterns differed by class: annotators agreed unanimously on 89 % (235/263) of discard cases, but for keep cases unanimous agreement dropped to 73 % (27/37). This asymmetry likely reflects the challenge of identifying non-standard, marginal meanings.

4.2 LLM annotation performance

The human consensus was the basis for evaluating LLM annotation performance. Table 1 offers a summary of the results in terms of precision, recall, F_1, MCC, and F_2, with models ranked by F_2 score due to its emphasis on recall (see Section 3.5).

According to F_2, the best performing model is Opus 4.5 in the pretraining condition (F_2 = 0.944), followed by the same model without pretraining (F_2 = 0.869). Both Opus 4.5 configurations achieved perfect recall, capturing all 37 keep cases in the dataset. Opus 4 also performed well, particularly with pretraining (F_2 = 0.857). Sonnet models occupy the lower end of the F_2 ranking, with Version 4.5 outperforming Version 4. The recall column shows that F_2 accurately captures recall performance, the F_2 ranking being largely equivalent to the recall ranking.

The improvement from Version 4 to Version 4.5 is substantial across both model families. Opus 4.5 pretrained achieves the highest scores on F_1 (0.871) and MCC (0.859) as well as F_2, meaning that all metrics select the same model. However, metric choice remains consequential for other configurations: F_1 and MCC would select Sonnet 4 basic (F_1 = 0.767, MCC = 0.768) over Opus 4.5 basic (F_1 = 0.725, MCC = 0.713), despite the latter achieving perfect recall. For a filtering task, Opus 4.5 basic is clearly preferable, but only F_2 captures this.

As for the impact of pretraining, the results are mixed. For Opus, examining the pretraining dataset improved performance in Version 4 (F_2 increased from 0.791 to 0.857) and also in Version 4.5, where it raised precision from 0.569 to 0.771 while maintaining perfect recall. For Sonnet, however, pretraining had a negative effect in both versions: F_2 dropped from 0.673 to 0.647 in Version 4 and from 0.808 to 0.761 in Version 4.5.

The types of error produced by each model are markedly different. Opus 4.5 produced no false negatives in either condition, but yielded 28 false positives without pretraining and 11 with pretraining. Opus 4 showed a similar pattern, with few false negatives (6 basic, 1 pretrained) but more false positives (17 basic, 26 pretrained). Sonnet 4, by contrast, produced no false positives in either condition but yielded 14 (basic) and 15 (pretrained)

Table 1: LLM performance against human consensus. Models sorted by F_2 score.

Model	Precision	Recall	F_1	MCC	F_2
Opus 4.5 (pretrained)	0.771	1.000	0.871	0.859	0.944
Opus 4.5 (basic)	0.569	1.000	0.725	0.713	0.869
Opus 4 (pretrained)	0.581	0.973	0.727	0.710	0.857
Sonnet 4.5 (basic)	0.640	0.865	0.736	0.703	0.808
Opus 4 (basic)	0.646	0.838	0.729	0.694	0.791
Sonnet 4.5 (pretrained)	0.612	0.811	0.698	0.657	0.761
Sonnet 4 (basic)	1.000	0.622	0.767	0.768	0.673
Sonnet 4 (pretrained)	1.000	0.595	0.746	0.750	0.647

false negatives. Sonnet 4.5 shows moderate numbers of both false negatives (5 basic, 7 pretrained) and false positives (18 basic, 19 pretrained). Figure 1 helps visualize the main patterns in the data.

Opus models tend to produce false positives rather than false negatives, prioritizing recall at the cost of precision. Sonnet 4 shows the opposite pattern, while Sonnet 4.5 occupies a middle ground. The improvement from Version 4 to 4.5 is visible in both families.

4.3 Efficiency analysis

This section examines the practical implications of model selection in this CL filtering task. Relying on the model with the highest F_2 score (Opus 4.5 pretrained) to filter the 300 instances in the testing dataset would mean reviewing 48 examples, that is, 16 % of the original dataset. Of the reviewed examples, 37 are potential cases of the target phenomenon and the remaining 11 are false positives. This approach ensures no information loss, as the model captures 100 % (37/37) of keep cases identified by humans, at the cost of examining some irrelevant examples (11/48). Conversely, before Version 4.5 became available, a researcher selecting the model with the highest F_1 or MCC score (Sonnet 4 basic) would have to review only 23 potential cases, with no time spent examining irrelevant examples. Nevertheless, since this model's coverage is only 62.2 %, 14 potential cases of the phenomenon would be lost.

The differences between the two approaches become more obvious when extrapolated to the entire dataset. Assuming the distribution remains constant throughout the 6,293 instances of the full corpus, in the first scenario (Opus 4.5 pretrained) the researcher would have to review 1,007 examples to capture 776 potential cases of the phenomenon, with no information loss. In the second scenario (Sonnet 4 basic), the researcher would only review 483 potential examples with no time spent examining false positives, but 294 cases would be lost. In terms of time efficiency, the first scenario results in a workload reduction of 84 %, with 5,286 instances avoided for manual review, resulting in around 5 h and 52 min needed to complete manual validation. The second scenario reduces

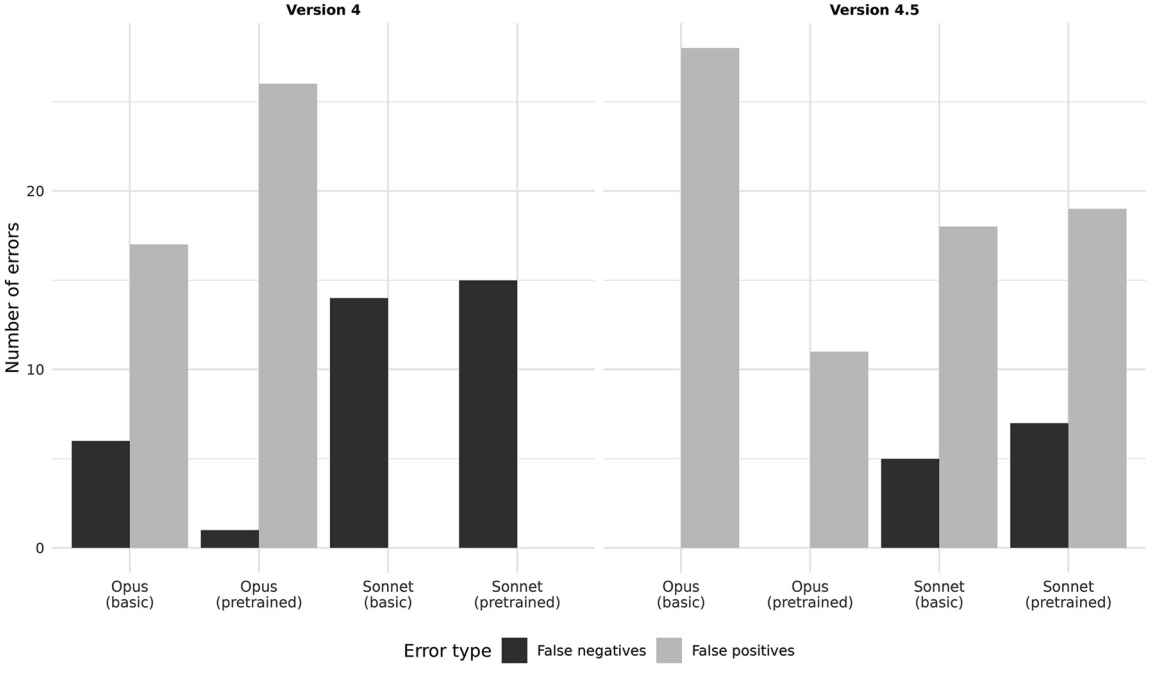

Figure 1: Error distribution by model and version.

the workload by 92.3 %, avoiding 5,810 cases and requiring a time investment of 2 h and 49 min. Both scenarios represent substantial time savings when compared to the 36 h and 43 min needed to review the whole dataset.[2]

5 Discussion and concluding remarks

This paper explored whether LLMs are effective tools to automate the annotation of corpus data. Specifically, it featured a CL filtering task consisting in the semantic disambiguation of the Galician word *pobo* by three human annotators and four LLMs (Claude Sonnet and Claude Opus in Versions 4 and 4.5). In contrast with previous contributions to the CL literature, this paper assumed that false negatives are much more costly at the filtering stage than false positives, as they may compromise the validity of the results: since discarded examples are not to be reviewed, a wrongly discarded example would be permanently lost for the analysis. This influenced methodological decisions regarding prompt design and validation metrics selection. Thus, prompts included explicit instructions to prioritize recall over precision in order to avoid false negatives, whereas model validation relied on F_2, which weighs recall more heavily than precision.

The findings can be summarized in terms of the research questions posed in Section 2:

(1) *How reliable are LLM annotations compared to manually-produced ones?* LLM annotations proved highly reliable, with the best model (Opus 4.5 pretrained) achieving perfect recall against the human consensus.
(2) *What are adequate metrics for validating LLM-based annotations in CL filtering tasks?* F_2 proved adequate for filtering tasks, as it prioritizes recall and correctly identifies models that minimize information loss.
(3) *Do larger, more computationally expensive LLMs perform better than smaller, more efficient ones in semantic annotation tasks?* Yes, within each version, larger models (Opus) consistently outperformed smaller ones (Sonnet) in terms of recall.
(4) *Does pretraining improve LLM annotation performance?* Pretraining improved performance for Opus but not for Sonnet, suggesting its benefits depend on model capacity.
(5) *What time efficiency gains can be obtained by using LLMs as first-pass filters in CL workflows compared to fully manual annotation?* Using the best-performing model reduces workload by 84 %, translating to substantial time savings.

Opus 4.5 showed strong performance against the human consensus, achieving perfect recall in both conditions and the highest F_2 score with pretraining (F_2 = 0.944). This is within the range of recent studies (Morin and Marttinen Larsson 2025a, 2025b; Yu et al. 2024), but comparability is limited due to differences in terms of task design and metric selection. Importantly, semantic disambiguation is inherently more challenging than the analysis of form-based patterns: even humans dealing with a polysemous word like *pobo* did not agree unanimously on 12.7 % of cases. In this context, Opus 4.5's perfect recall ensures no information loss, capturing all 37 cases of the target meaning in a dataset of 300 examples.

Opus showed great understanding of the task goals, especially in relation to the priority of recall over precision. Sonnet received the same instructions, but results varied by version. Sonnet 4 performed poorly in terms of recall, producing many false negatives and no false positives, suggesting it failed to grasp the priority given to recall. Sonnet 4.5 showed a pattern more similar to Opus, with fewer false negatives and more false positives, indicating better comprehension of task priorities. However, pretraining had a negative effect on Sonnet in both versions, unlike Opus, where it consistently improved performance. This suggests that the benefits of pretraining depend on model capacity: more powerful models like Opus leverage pretraining examples to refine their understanding, while less powerful models may be confused by them.

In this context, establishing an evaluation framework consistent with research goals is critical. Following NLP practices, previous CL studies have relied on F_1 and MCC to assess LLM performance. However, these metrics do not align with the objectives of filtering tasks, where the risks of false negatives far outweigh the costs of false

2 Time estimations assume that the binary classification of examples requires the same investment of human time as the binary validation of LLM-generated results (see the end of Section 3.2).

positives. In our study, while all metrics select the best overall model (Opus 4.5 pretrained), F_1 and MCC would rank Sonnet 4 (basic) above Opus 4.5 (basic), despite the latter achieving perfect recall. By contrast, F_2 consistently identifies the models that prioritize recall, aligning with the goals of a filtering task.

Emphasis on recall comes at the cost of poorer precision performance. Yet, this trade-off still yields significant efficiency gains. Using Opus 4.5 pretrained, manual review of examples is cut down to 16 % of the original sample, with no information lost. When extrapolated to the entire dataset, this workload reduction materializes in time savings of more than 30 h.

These considerations lead us to conclude that LLMs are viable first-pass filters for semantic disambiguation tasks in CL. However, using LLMs does not by itself ensure successful results, and linguists must be aware of a number of methodological practices that help to enhance LLM performance in at least two key areas: prompt design and model validation. While the importance of the latter has already been discussed, prompt design is another crucial step of an LLM-assisted CL workflow. A dynamic, iterative approach to prompt design has been proposed, but current LLM memory limitations favor a static, single-phase strategy. Following a static approach, the researcher would apply the same prompt across different conversations, ensuring consistency, scalability, and reproducibility, at the cost of upfront investment in prompt design. This requires acquiring solid prior understanding of the phenomenon under investigation, and will usually involve several rounds of trial-and-error refinement. Additionally, as shown in this work and previous ones, especially Morin and Marttinen Larsson (2025a), researchers must pay close attention to detail, ensuring correct formal layout, grammatical correctness, terminological precision, and explicitness. The use of XML tags has proven especially fruitful, allowing for clear structuring and unambiguous cross-referencing. Moreover, asking the model to think and include its reasoning in the output significantly improves performance. The fact that the same prompt was applied without modification to both Version 4 and Version 4.5 models, with improved results in the latter, demonstrates the transferability of a well-designed static prompt across model updates. Recent work confirms that these prompt engineering principles can scale to very large datasets when implemented via API batch processing (Morin and Marttinen Larsson 2025b), though such approaches require programming expertise not assumed in the present study.

Before LLMs can be confidently integrated into standard CL practice, more experimental applications diving into a wider range of languages, phenomena, and methodological processes are needed. In this regard, it is important to acknowledge the limitations of the present study and identify the main areas requiring further investigation. We have focused on a semantic disambiguation task in Galician. This contributes new perspectives to the increasing body of LLM applications by showing both that automatic semantic annotation of corpus data is viable and that low-resource languages can benefit from LLM capabilities. However, the scope of this study was limited to the analysis of a single, polysemous word, and transferability to other, more complex semantic phenomena is not guaranteed. Similarly, Galician, being a low-resource language, is part of the major Romance linguistic family and is formally very close to high-resource languages like Portuguese and Spanish. Thus, cross-linguistic applicability of LLM-assisted workflows to a typologically diverse set of languages is an outstanding challenge. From a methodological perspective, the use-case of a CL filter illustrates only one possible application of LLMs to corpus annotation. Research into the coding of predictor variables and the challenges it poses for LLM-assisted workflows would be very welcome. Finally, this study relied on four LLMs from the same provider. Future research would benefit from comparing a wider range of LLMs across different providers, so as to find the best performing tools for different tasks. Additionally, the present study annotated a validation sample of 300 examples using a web-based interface, prioritizing accessibility over scale. Once validated, this method can be applied to the full dataset of 6,293 instances. For even larger datasets, API-based approaches, like that by Morin and Marttinen Larsson (2025b), demonstrate that tens of thousands of examples can be processed automatically, representing a complementary strategy for researchers with programming expertise.

The use of commercial LLMs for corpus annotation raises ethical and practical considerations. In this study, data came from CORGA, a publicly accessible corpus whose underlying texts are copyrighted. Use of concordances is permitted under citation rights, and the limited context windows provided by the corpus interface ensure compliance with copyright restrictions. Additionally, metadata identifying sources was omitted from the files shared with the LLM, and no personal or sensitive information was involved. Researchers working with restricted corpora or sensitive data should take precautions to prevent corpus materials from entering model

training datasets. Both API access and web-based interfaces offer options to disable data sharing: for instance, Morin and Marttinen Larsson (2025b) conducted all API processing with data sharing disabled to comply with corpus licensing restrictions, while some web interfaces offer an incognito mode that ensures that conversations are not used to train models. As LLM-assisted annotation becomes more widespread, the field should develop explicit guidelines addressing copyright, data protection, reproducibility, and transparency in reporting LLM use.

The growing use of LLMs in CL will have serious implications for the field at large, and will likely change annotation practices drastically. This shift requires transparent methodological frameworks that balance automation with human oversight to maintain accountability and scholarly rigor. As LLM-assisted corpus annotation moves from experimental application to standard practice, the field will require new training protocols for researchers and updated reproducibility guidelines. The findings presented here aim to contribute to the responsible integration of these powerful new tools into the CL workflow.

Acknowledgments: I am very grateful to Maruxa Álvarez and Francisco Dubert for generously volunteering their time to annotate the linguistic examples used in this study. I also want to thank two anonymous reviewers for their very helpful suggestions, which have greatly improved the initial version of this paper.
Funding information: This work was funded by the Spanish Ministry of Science, Innovation and Universities (MICIU) and the State Research Agency (AEI) under grant PID2022-137170OB-I00 (10.13039/501100011033), and by the European Regional Development Fund (ERDF/EU).
Author contributions: The author confirms the sole responsibility for the conception of the study, presented results and manuscript preparation.
Conflict of interest: The author states no conflict of interest.
Data availability statement: The prompts, annotated datasets, and other supplementary materials used in this study are available in the Open Science Framework repository at https://doi.org/10.17605/OSF.IO/DJA3K.
Declaration of generative AI in the writing process: During the preparation of this work the author used Claude 4 Sonnet to improve the readability and clarity of the first draft of the paper. All research design, data analysis, interpretation of results, and core arguments are entirely the author's original work. The author takes full responsibility for the content of the published article.

References

Anthropic. 2025a. *Claude* (Version 4). https://claude.ai/.
Anthropic. 2025b. *Claude* (Version 4.5). https://claude.ai/.
Anthropic. 2025c. *Introducing Claude 4*. https://www.anthropic.com/news/claude-4.
Artstein, R. 2017. Inter-annotator agreement. In N. Ide & J. Pustejovsky (eds.), *Handbook of linguistic annotation*, 297–313. Dordrecht: Springer.
Burkow, J., G. Holste, J. Otjen, F. Perez, J. Junewick, A. Zbojniewicz, E. Romberg, S. Menashe, J. Frost & A. Alessio. 2024. High sensitivity methods for automated rib fracture detection in pediatric radiographs. *Scientific Reports* 14(1). 8372.
Centro Ramón Piñeiro para a investigación en humanidades. 2024. *Corpus de Referencia do Galego Actual (CORGA)* (Version 4.1). https://corpus.cirp.gal/corga/.
Chicco, D., N. Tötsch & G. Jurman. 2021. The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData Mining* 14(1). 13.
Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* 20. 37–46.
Fleiss, J. L. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin* 76(5). 378–382.
Gilardi, F., M. Alizadeh & M. Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences* 120(30). https://doi.org/10.1073/pnas.2305016120.
González González, M. N.d. *Dicionario da Real Academia Galega*. Real Academia Galega. https://academia.gal/dicionario/.
Gries, S. T. 2025. On regression modeling in varieties research. *World Englishes* 44(1–2). 57–77.
Ispahany, J., M. R. Islam, M. A. Khan & M. Z. Islam. 2025. iCNN-LSTM+: A batch-based incremental ransomware detection System using sysmon. *IEEE Access* 13. 87978–87998.
Kortmann, B. 2021. Reflecting on the quantitative turn in linguistics. *Linguistics* 59(5). 1207–1226.
Landis, J. R. & G. G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics* 33(1). 159–174.
Manning, C. D., P. Raghavan & H. Schütze. 2008. *Introduction to information retrieval*. Cambridge: Cambridge University Press.

Mariño Paz, R. 2024. O uso de *pobo* coa acepción de 'entidade de poboación' na historia da lingua galega. *Revista de Filología Románica* 41. 59–74.

Mattera, G., J. Polden & J. Norrish. 2024. Monitoring the gas metal arc additive manufacturing process using unsupervised machine learning. *Welding in the World* 68(11). 2853–2867.

Morin, C. & M. Marttinen Larsson. 2025a. Large corpora and large language models: A replicable method for automating grammatical annotation. *Linguistics Vanguard* 11. 501–510.

Morin, C. & M. Marttinen Larsson. 2025b. A large-scale, unsupervised pipeline for automatic corpus annotation using LLMs: Variation and change in the English consider construction. arXiv: 2510.12306 [cs]. https://doi.org/10.48550/arXiv.2510.12306.

Newman, J. & C. Cox. 2020. Corpus annotation. In M. Paquot & S. T. Gries (eds.), *A practical handbook of corpus linguistics*, 25–48. Cham: Springer.

Stefanowitsch, A. 2020. Corpus linguistics: A guide to the methodology. *Language Science Press*. https://doi.org/10.5281/zenodo.3735822.

Tessema, B. M., A. Kedia & T.-S. Chung. 2024. UnifiedCrawl: Aggregated common crawl for affordable adaptation of LLMs on low-resource languages. arXiv: 2411.14343 [cs]. https://doi.org/10.48550/arXiv.2411.14343.

Torrent, T. T., T. Hoffmann, A. L. Almeida & M. Turner. 2024. *Copilots for linguists: AI, constructions, and frames*. Cambridge: Cambridge University Press.

Uhm, M., J. Kim & G. Lee. 2025. Automated analysis of construction safety accident videos using a large multimodal model and graph retrieval-augmented generation. *Automation in Construction* 177. 106363.

Wei, J., X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le & D. Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Proceedings of the 36th international conference on neural information processing systems*, 24824–24837. Red Hook, NY: Curran Associates Inc.

Yae, J. H., N. C. Skelly, N. C. Ranly & P. M. LaCasse. 2025. Leveraging large language models for word sense disambiguation. *Neural Computing and Applications* 37(6). 4093–4110.

Yu, D., L. Li, H. Su & M. Fuoli. 2024. Assessing the potential of LLM-assisted annotation for corpus-based pragmatics and discourse analysis: The case of apology. *International Journal of Corpus Linguistics* 29(4). 534–561.

Zhao, X., Y. Liu & Q. Zhao. 2024. Improved LightGBM for extremely imbalanced data and application to credit card fraud detection. *IEEE Access* 12. 159316–159335.

Zhong, T., Z. Yang, Z. Liu, R. Zhang, Y. Liu, H. Sun, Y. Pan, Y. Li, Y. Zhou, H. Jiang, J. Chen & T. Liu. 2024. Opportunities and challenges of large language models for low-resource languages in humanities research. arXiv:2412.04497 [cs]. https://doi.org/10.48550/arXiv.2412.04497.