Lost in Speech: Benchmarking, Evaluation, and Parsing of Spoken Bilingual Conversational Language Beyond Standard UD Assumptions

Anonymous ACL submission

Abstract

Spoken bilingual conversations pose substantial challenges for syntactic parsing because they often include disfluencies and discourse-driven structures that complicate dependency parsing under standard Universal Dependencies (UD) assumptions and evaluation practices. To systematically study these challenges, in this work, we first introduce a linguistically grounded taxonomy of conversational bilingual phenomena, together with **SpokeBench**, an expert-annotated English-Spanish benchmark for structurally complex speech. To address the limitations of existing evaluation practices, we propose **Flex-UD**, an ambiguity-aware evaluation metric that distinguishes catastrophic structural failures from linguistically acceptable variations. Finally, we introduce **DECAP**, a decoupled agentic parsing framework that separates spoken-phenomena handling from core syntactic analysis, enabling robust and interpretable dependency parsing without retraining. Experiments across both proprietary and open-weight LLMs show that **DECAP** substantially improves performance on complex conversational phenomena and achieves over **60%** improvements in UPOS-F1 Score over baselines, while Flex-UD evaluations reveal gains that otherwise remain partially hidden under standard attachment-based metrics[1].

1 Introduction

Unlike written text, conversational speech frequently contains disfluencies and discourse-driven structures that violate many of the assumptions underlying canonical annotation frameworks such as Universal Dependencies (UD). These phenomena are particularly common in bilingual interaction, where speakers dynamically negotiate structure across languages during code-switching. As illustrated in Figure 1, models that perform well on written text often fail on conversational bilingual transcripts, producing brittle or linguistically implausible analyses when faced with spoken-language variability and structural ambiguity.

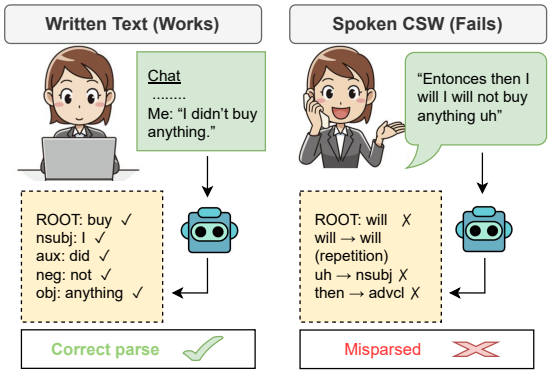

Figure 1: Illustration of the modality gap motivating this work. Parsers and LLMs typically produce well-formed dependency analyses for written text (left), but the same systems often misparse spoken bilingual conversational utterances (right) due to disfluencies, repetition, and discourse phenomena that violate written-text assumptions.

Recent work has shown that large language models (LLMs) can serve as effective tools for syntactic annotation and parsing in multilingual and low-resource contexts, particularly for written text (Tian et al., 2024; Kellert et al., 2025; James et al., 2022; Huzaifah et al., 2024). However, their performance remains understudied and inconsistent for conversational bilingual transcripts, particularly in spoken-language contexts, which introduce ambiguity that is often poorly captured by standard parsing evaluation practices. This degradation cannot be explained by bilingualism or data scarcity alone. Rather, it reflects a mismatch between written-language annotation conventions, rigid evaluation practices, and the incremental, interactional nature of conversational speech (Kahane et al., 2021; Dobrovoljc, 2022). As a result, structurally plausible analyses are often penalized as outright parsing

[1]Data and source code are available at https://anonymous.4open.science/r/LISAE78

failures. Many parsing failures arise from limited spoken training data and structural assumptions, including non-canonical roots and discourse-level attachments, embedded within existing annotation schemes and evaluation metrics. This suggests that improvements in model scale alone are unlikely to provide a robust or language-agnostic solution.

In this work, we argue that progress in parsing spoken bilingual conversational language requires jointly rethinking annotation, evaluation, and parsing abstractions rather than treating conversational phenomena as noise around written-language syntax. We adopt a systems-oriented perspective that explicitly separates spoken-language phenomena from core syntactic analysis, grounding this design in an empirical study of the structures that consistently undermine canonical parsing assumptions. Building on this analysis, we introduce new resources, evaluation strategies, and parsing abstractions designed specifically for structurally variable conversational English-Spanish speech. Our contributions are:

- We design a linguistically grounded **taxonomy** of spoken bilingual conversational phenomena, distinguishing spoken-specific structures from broader non-canonical constructions and analyzing their structural and cognitive complexity through a **human annotation survey** measuring annotation difficulty, ambiguity, and cognitive load. These human judgments provide an empirical basis for evaluating and interpreting automatic parsing performance.
- We present **SpokeBench**, an expert-annotated and adjudicated gold benchmark for evaluating dependency parsing on structurally complex bilingual conversational transcripts.
- We introduce **DECAP**, a decoupled agentic parsing framework that improves robustness and interpretability by isolating spoken-language handling from core syntax.
- We propose **Flex-UD**, a weighted, ambiguity-aware evaluation metric that penalizes structurally catastrophic errors more than acceptable variation.

Together, these contributions provide a foundation for studying and evaluating syntactic parsing beyond written-language assumptions, highlighting the importance of ambiguity-aware evaluation and speech-informed parsing abstractions for conversational bilingual language.

2 Related Work

UD Parsing for Non-Canonical and Code-Switched Text. UD is a widely adopted framework for cross-linguistic syntactic annotation (de Marneffe et al., 2021; Nivre et al., 2016), but its application to non-canonical data such as conversational speech and intra-sentential bilingual code-switching remains problematic (Kellert et al., 2025). Spoken and code-switched utterances frequently violate core UD assumptions, including clause completeness and canonical head selection, leading to substantial annotation ambiguity (Çetinoğlu and Çöltekin, 2019; Kellert et al., 2025; Bhat et al., 2018). To address these challenges, researchers have developed dedicated bilingual treebanks (Çetinoğlu and Çöltekin, 2023; Kyle et al., 2022; Pulido et al., 2025; Sanguinetti et al., 2023) and explored alternative dependency formalisms such as Surface-Syntactic Universal Dependencies (SUD), which modify head assignment to better reflect spoken structure (Kahane et al., 2021; Gerdes et al., 2019b). Despite these efforts, prior work has largely focused on documenting annotation difficulties rather than enabling robust, scalable parsing or evaluation of spoken bilingual data. As a result, current models remain limited in their ability to support accurate syntactic analysis, large-scale computational processing, and systematic evaluation of spoken bilingual language.

LLMs for Annotation and Parsing. Large Language Models (LLMs) have recently been explored as parsers and annotation tools, showing that they can produce syntactic analyses for written text in zero-shot or few-shot settings (Zhang et al., 2025), and achieve competitive performance when fine-tuned (Bai et al., 2025). These capabilities motivate their use in bootstrapping annotations and generating synthetic treebanks, even for low-resource languages. However, consistent limitations remain. Beyond the well-documented challenges in written language, studies indicate that LLM performance degrades significantly with flexible, noisy, or conversational input, including spoken and CSW data (De Leon et al., 2024). Prior works have explored alternative parsing architectures, such as sequence-labeling approaches (Gómez-Rodríguez et al., 2024; Imran et al., 2026a) or incorporating syntactic information into downstream tasks (Imran et al., 2026b). While these models show promise for certain spoken-language tasks (e.g., detection or translation) (James et al., 2022; Huzaifah et al.,

Category	Category Definition	Example
	Phenomena Shared Across Written and Spoken Conversation	
Contractions	Tokens encoding multiple syntactic units, such as English contractions or Spanish merged forms, which require explicit segmentation during dependency annotation.	we're *[we are]*, al *[to the]*
Compound / MWE	Multi-word expressions and conventionalized compounds functioning as a single syntactic unit that may obscure internal dependency relations when treated compositionally.	F_bat, spring_break
Slang / Curses	Pragmatically dependent expressions whose syntactic interpretation varies substantially across discourse and cultural context, often resulting in inconsistent annotation and parser behavior.	shit, papi *[daddy]*
Enclisis	Spanish verbal constructions containing attached clitic pronouns that require morpheme-level decomposition for accurate dependency assignment.	fíjat*e [(you) look]*, convirtiéndo*se [making itself]*
	Spoken-Specific Conversational Phenomena	
Repetition	Recurrence of words or phrases during incremental speech production, often used to maintain the conversational floor while planning upcoming material. We observe (a.) single-token repetition, (b.) token-separated repetition, and (c.) code-switched repetition across languages.	(a.) es es *[it's it's]*, (b.) casa uh casa *[house uh house]*; (c.) short cortos *[short short]*
Discourse Elements	Pragmatic markers, interjections, and conversational connectives whose syntactic integration varies across contexts. We distinguish between syntactically integrated discourse elements and pragmatically motivated insertions with weak structural attachment.	I mean yo tengo *[I mean I have]*
Ellipsis	Structurally incomplete utterances arising from interruption, abandonment, or incremental production. Elliptical constructions frequently omit expected syntactic arguments, complicating dependency assignment and recoverability.	ah before we; yeah ella se *[yeah she]*
False Start	Mid-utterance reformulations in which a speaker abandons an initiated structure and replaces it with a revised continuation, often reflecting online adjustment of bilingual grammatical planning and requiring inference over partially realized syntactic structure and speaker intent.	pero I lo I use it *[but I it I use it]*
Filler Words	Floor-holding vocalizations carrying minimal semantic content but remaining interactionally important in conversational speech, often with weak syntactic integration.	uh, um, eh

Table 1: Taxonomy of conversational bilingual phenomena identified in the Miami Corpus. Phenomena are grouped by whether they occur in both written and spoken discourse or are primarily associated with conversational speech.

2024), ability to robustly parse fragmented or ambiguous syntax of real speech remains inconsistent.

Evaluation Limitations. Standard dependency parsing metrics such as Labeled and Unlabeled Attachment Score (LAS and UAS) rely on strict matching against a single gold tree, making them poorly suited to conversational and structurally ambiguous language where multiple analyses may be linguistically plausible. Prior work has shown that these metrics often conflate genuine parsing failures with acceptable structural variation (Sterner and Teufel, 2025b,a), motivating the need for ambiguity-aware evaluation.

3 Spoken Bilingual Conversation: A Non-Canonical Parsing Domain

In this section, we examine spoken bilingual conversational text as a complex parsing domain. First, we develop a linguistically grounded taxonomy of conversational phenomena, followed by a human-perception survey investigating their structural and cognitive complexity. We then introduce **SpokeBench**, a gold benchmark for evaluating parsing performance under spoken-text conditions.

3.1 Taxonomy of Spoken Bilingual Conversational Phenomena

To identify the primary sources of parsing difficulty in conversational bilingual language, we conducted a data-driven analysis of English-Spanish utterances from the Miami Corpus (Deuchar et al., 2014), a widely used English-Spanish bilingual conversational dataset. Approximately 2,800 sentences were independently examined by trained linguists, who analyzed and identified conversational phenomena that complicate non-canonical dependency parsing. The consolidation of these observations yielded a linguistically grounded taxonomy, motivated by recurrent parser failures and annotation difficulties. Many of these phenomena can in principle be represented within existing UD and SUD frameworks, but conversational bilingual speech substantially increases annotation ambiguity and structural variability in practice. Our goal is therefore not to replace existing annotation schemes, but to systematically analyze how these conversational phenomena interact with bilingual speech and affect parser behavior, annotation consistency, and evaluation.

We organize the identified phenomena into two groups: (i) phenomena occurring in both written and spoken discourse but becoming structurally more challenging in bilingual conversational settings, and (ii) phenomena strongly associated with spontaneous conversational speech and incremental language production. The definitions and examples are provided in Table 1.

3.2 SpokeBench: A Gold Benchmark for Spoken Conversational Parsing

Subset Selection Strategy. Based on the conversational phenomena identified in the Miami Corpus, we construct **SpokeBench**, a gold benchmark for evaluating dependency parsers on structurally complex bilingual conversational transcripts. Since annotating the full set of over 2,800 sentences was infeasible due to the cost and ambiguity of conversational speech annotation, we selected a curated subset balancing coverage of spoken phenomena and structural complexity. Most of the sentences have overlapping phenomena and some categories are extremely rare, making it hard to strictly isolate sentences for each category in the taxonomy. Therefore, we constructed SpokeBench with a balance of isolated dominant categories (Repetition, Contraction [EN & ES], Ellipsis, Discourse, Fillers) and categories (denoted with '+') that consist of a dominant phenomenon along with 1 or 2 rare phenomena, such as compound, slang, or false start. The category "Complex" refers to the sentences that have 3+ phenomena and "Others" refer to sentences unique in their presentation but do not fit in the above categories. The resulting benchmark contains 160 sentences spanning ten categories; its composition is summarized in Appendix A. Unlike most existing UD treebanks, which primarily focus on annotation coverage or multilingual representation, SpokeBench is explicitly designed to evaluate parser robustness under structurally ambiguous conversational conditions.

Annotation Procedure. The Annotations in **SpokeBench** follow a modified version of UD guidelines adapted for spoken bilingual conversational data. Standard UD conventions were applied where possible, with additional rules introduced for repetitions, contractions, discourse markers, fillers, and ellipsis. Each sentence was independently annotated by at least two linguistically trained annotators (seven in total), and disagreements were evaluated using a graded acceptability protocol distinguishing acceptable variation from substantive annotation errors. Borderline or unacceptable cases were resolved through expert adjudication, yielding a consensus gold annotation for each sentence. To evaluate annotation consistency, we calculated Cohen's Kappa score for a randomized subset of 50 sentences, which resulted in 0.798. Appendix C contains the detailed annotation guidelines and review procedures.

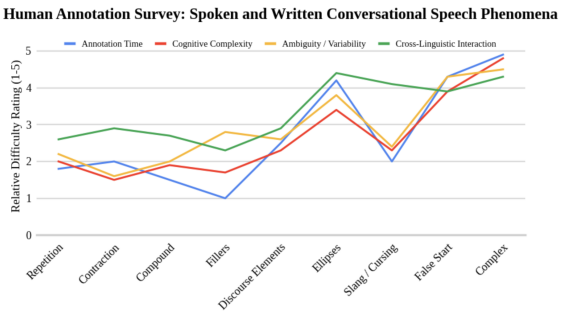

Figure 2: Average Likert-scale ratings from expert annotators across four dimensions: annotation time, cognitive complexity, ambiguity/variability, and cross-linguistic interaction.

3.3 Human Perception Survey

Following the annotation process, we conducted a human perception survey involving the expert annotators who participated in the construction and adjudication of **SpokeBench**. Annotators evaluated each conversational phenomenon on a five-point Likert scale (1 = least difficult, 5 = most difficult) across four dimensions: annotation time, cognitive complexity, ambiguity/variability, and cross-linguistic interaction. As shown in Figure 2, contractions, filler words, and localized repetitions were generally perceived as less difficult to annotate, while ellipsis, discourse elements, and false-start constructions received the highest difficulty ratings across nearly all dimensions due to their reliance on contextual interpretation, structural recoverability, and speaker-intent inference. These findings suggest that conversational phenomena involving greater ambiguity and discourse-level reasoning present substantially greater challenges for human annotators. We hypothesize that the same phenomena will also present systematic challenges for automatic parsers, particularly for dependency attachment and structural recoverability. This hypothesis is tested experimentally in Section 5.

4 DECAP: A Decoupled Agentic Framework for Spoken Bilingual Conversation Parsing

In this section, we introduce **DECAP** (DECoupled Agentic Parser), a modular framework for parsing conversational bilingual transcripts. Rather than treating conversational variability within a single monolithic parser, DECAP separates spoken-phenomena handling, language-specific normalization, syntactic structure assignment, and global val-

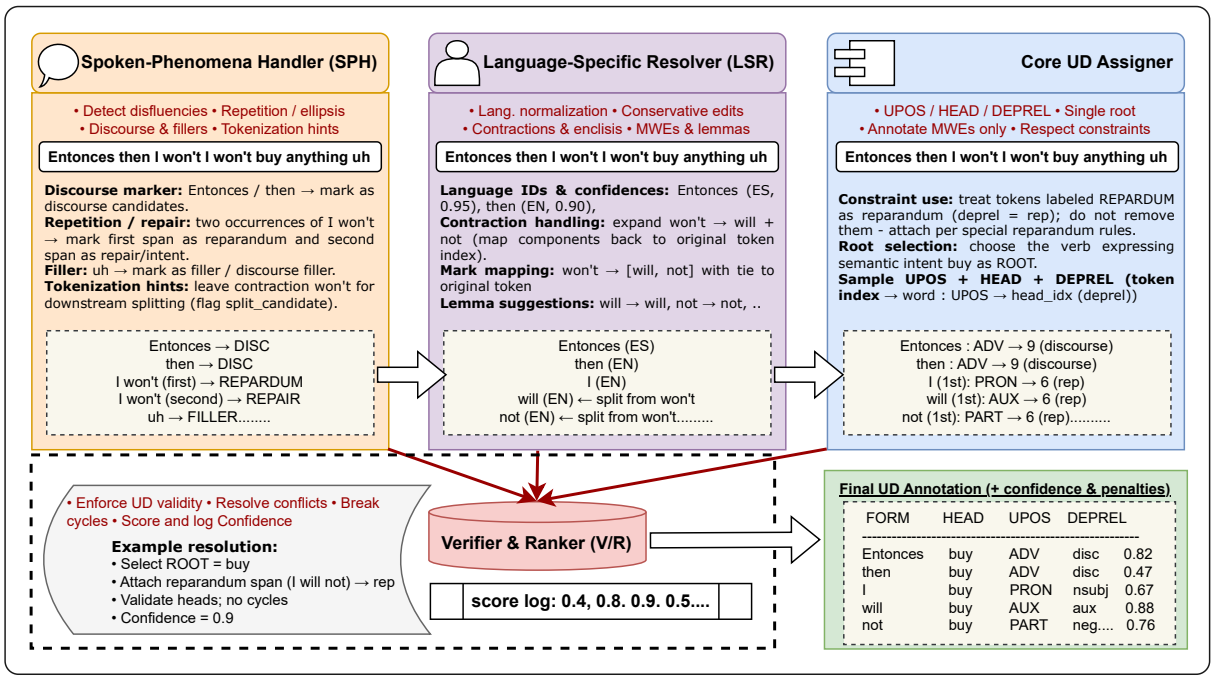

Figure 3: Overview of the **DECAP** framework for spoken code-switching parsing, illustrated with a running example. The input utterance ("*Entonces then I won't I won't buy anything uh*") is processed. The **Spoken-Phenomena Handler (SPH)** detects disfluencies (e.g., repetition, discourse markers, fillers) and provides tokenization hints. The **Language-Specific Resolver (LSR)** applies conservative, language-aware normalization, and contraction splitting. The **Core UD Assigner** constructs a dependency parse under these constraints, preserving reparanda and enforcing a single root. Finally, the **Verifier and Ranker (V/R)** enforces global UD validity and outputs a final annotation with confidence and penalty scores, demonstrating incremental resolution of ambiguity.

idation into specialized stages. The framework is guided by three principles: (i) *instruction grounding*, leveraging LLMs' ability to follow explicit linguistic constraints; (ii) *modularity*, isolating distinct sources of conversational complexity; and (iii) *extensibility*, enabling adaptation across languages, phenomena, and LLM families without retraining. Figure 3 illustrates the overall architecture.

Spoken-Phenomena Handler (SPH). The SPH agent identifies conversational phenomena prior to syntactic parsing, including repetitions, ellipsis, discourse markers, fillers, contractions, and MWEs, and proposes minimal tokenization edits where required. Rather than directly assigning dependency relations, SPH provides lightweight structural guidance and tokenization hints that constrain downstream parsing decisions and reduce error propagation across later stages.

Language-Specific Resolver (LSR). This agent applies conservative language-aware normalization to the output of SPH, including contraction expansion, multiword-expression validation, and lemma normalization. The agent is designed to minimize unnecessary structural repairs while preserving conversational structure and English-Spanish language variations. By isolating language-dependent operations, LSR improves robustness across multilingual and code-switched input.

Core UD Structure Assigner. The Core UD Assigner produces dependency annotations over the normalized representation generated by previous agents. It assigns UPOS tags, dependency heads, and dependency relations while enforcing standard UD structural constraints such as single-root validity. Conversational phenomena identified by earlier stages are preserved rather than reconstructed into canonical written-language forms. This stage focuses exclusively on syntactic structure, deferring validation and correction to the final agent.

Verifier and Ranker (V/R). The V/R agent integrates the outputs of all previous stages and performs global structural validation, resolving issues such as invalid heads, dependency cycles, and conflicting annotations. In addition to producing sheet-ready annotations, V/R assigns per-token confidence and penalty scores, yielding an interpretable record of uncertainty and intervention. This final stage guarantees that DECAP outputs

are structurally valid, auditable, and suitable for downstream evaluation.

Interaction protocol. DECAP applies a fixed, deterministic sequence of agents to each sentence, with each stage consuming the structured output of the previous one and enforcing increasingly global constraints, as detailed in the Appendix. D.

5 Experiments

5.1 Experimental Setup

Dataset. All experiments are conducted on **SpokeBench**, the benchmark introduced in Section 3. We evaluate systems on a curated subset of 160 sentences selected to balance conversational phenomena and structural complexity. Gold Universal Dependency annotations are used across all experiments to ensure controlled and directly comparable evaluation.

Models. We evaluate four parser configurations spanning both traditional and LLM-based approaches. As a traditional baseline, we use the **Trankit** parser, a transformer-based multilingual dependency parsing toolkit built on XLM-RoBERTa (Nguyen et al., 2021). It supports multilingual UD parsing through contextualized transformer representations and automatic language identification, but is trained on monolingual treebanks only. We then evaluate the proposed **DECAP** framework instantiated with three different LLM families: GPT-4.1 (version 2025-04-14), `meta-llama/Llama-3.1-8B-Instruct`, and `meta-llama/Llama-3.1-70B-Instruct`. All models are evaluated in a zero-shot setting with deterministic decoding (temperature = 0), and none are fine-tuned on **SpokeBench** or Miami Corpus. This setup allows us to evaluate the extent to which the DECAP framework generalizes across both proprietary and open-weight LLM architectures.

5.2 Evaluation Metrics

Standard UD Metrics and Limitations. We evaluate parser performance using four standard dependency parsing metrics: Unlabeled Attachment Score (UAS), Labeled Attachment Score (LAS), Content-Labeled Attachment Score (CLAS), and UPOS-F1 Score (UP-F1). UAS measures unlabeled head attachment accuracy, LAS additionally requires correct dependency labels, and CLAS evaluates labeled dependency accuracy over core content relations only. We additionally report UPOS-F1 to evaluate morphosyntactic tagging independently from dependency structure assignment. While these metrics are effective for canonical written text, they are less well suited to conversational bilingual speech, where disfluencies, ellipsis, discourse markers, and structurally incomplete utterances often permit multiple linguistically plausible analyses. As a result, strict gold-tree matching may penalize acceptable structural variation alongside genuine parsing errors.

Flex-UD Metric. To better account for structural ambiguity in conversational speech, we introduce **Flex-UD** (Flexible Evaluation for Universal Dependencies), a composite severity-aware evaluation metric designed for spoken bilingual parsing. Rather than relying exclusively on exact tree matching, Flex-UD evaluates parses across multiple complementary dimensions, including accurate tokenization, ID assignments, UPOS assignments, dependency heads, and dependency relations. The metric is designed to: (i) tolerate linguistically acceptable variation, (ii) distinguish catastrophic structural errors from minor deviations, and (iii) provide interpretable diagnostics for qualitative error analysis. Illustrative examples of error severity and penalty assignment are provided in Appendix E.

Formal Definition. The Flex-UD metric evaluates parses across five component scores $S = \{s_{\text{Split}}, s_{\text{ID}}, s_{\text{UPOS}}, s_{\text{HEAD}}, s_{\text{DEPREL}}\}$, corresponding respectively to tokenization accuracy, ID assignment, UPOS agreement, dependency head assignment, and dependency relation assignment. Each score is normalized to the range $[1, 100]$ and assigned a weight $w = \{0.10, 0.05, 0.15, 0.25, 0.45\}$, placing greater emphasis on dependency structure and relation accuracy. The weighted aggregate score is computed as

$$\text{raw} = \sum_i w_i s_i.$$

To account for severe structural failures, Flex-UD applies a bounded penalty term $P \in [0, 1]$, where larger values correspond to more severe parsing inconsistencies. Catastrophic errors such as invalid dependency heads, unresolved cycles, severe tokenization mismatches, or structurally incompatible alignments receive substantially larger penalties than minor linguistic variations such as discourse attachment ambiguity or near-equivalent dependency relations. The final Flex-UD score is

Category	TRANKIT Bilingual				DECAP Llama 3 8B				DECAP Llama 3 70B				DECAP GPT-4.1			
	LAS	UAS	CLAS	UP-F1	LAS	UAS	CLAS	UP-F1	LAS	UAS	CLAS	UP-F1	LAS	UAS	CLAS	UP-F1
Repetition	5.3	7.7	3.3	12.4	9.8	15.2	7.8	24.0	25.9	48.2	18.8	65.8	42.5	64.7	26.7	84.7
Repetition+	3.9	6.1	2.1	10.3	8.4	13.3	5.9	21.4	21.3	41.6	15.6	58.3	38.9	49.3	29.3	81.6
Contr. (EN)	4.2	7.4	1.3	8.7	1.6	4.6	0.3	28.6	21.5	37.7	11.8	57.3	31.7	48.2	25.5	83.5
Contr. (ES)	3.4	8.3	1.2	9.4	0.8	3.7	0.1	18.8	28.4	46.3	20.5	63.1	42.8	51.7	33.8	82.6
Ellipsis	1.7	4.3	0.3	13.7	10.4	14.6	7.5	29.3	34.8	50.1	22.9	61.8	33.6	43.6	27.1	81.4
Ellipsis+	0.4	2.1	0.0	10.6	7.3	11.5	3.1	17.9	31.5	48.3	17.9	55.8	45.7	59.4	26.5	81.9
Discourse	1.1	4.1	0.1	11.2	5.9	8.9	1.9	25.5	25.7	38.9	19.3	58.7	31.2	40.1	21.2	86.2
Discourse+	0.2	1.7	0.0	7.8	6.8	7.6	1.6	20.6	31.6	40.8	21.0	51.6	35.4	53.9	30.1	78.7
Complex	1.8	4.8	1.3	9.1	3.6	5.2	0.3	18.9	28.1	43.3	21.6	53.1	32.9	41.8	27.3	80.5
Fillers	0.0	0.3	0.0	2.3	1.6	3.4	1.2	15.8	30.3	47.9	26.9	59.2	31.6	38.7	17.8	77.8
Others	7.6	10.6	6.3	8.7	8.7	14.9	15.3	16.2	32.8	41.7	24.3	58.8	20.1	31.6	15.9	62.8
Overall	**2.7**	**5.2**	**1.6**	**9.4**	**6.2**	**9.7**	**4.5**	**20.8**	**28.6**	**43.8**	**20.1**	**57.6**	**35.0**	**47.4**	**25.6**	**79.2**

Table 2: Performance of parsers tested across spoken bilingual conversational categories under standard UD metrics. UP-F1 denotes UPOS-F1 Score

Category	TRANKIT Bilingual				DECAP Llama 3 8B				DECAP Llama 3 70B				DECAP GPT-4.1			
	UPOS	HEAD	DEP	Flex-UD	UPOS	HEAD	DEP	Flex-UD	UPOS	HEAD	DEP	Flex-UD	UPOS	HEAD	DEP	Flex-UD
Repetition	18.9	14.2	11.3	13.4	38.1	21.4	23.8	25.6	70.8	51.3	57.6	60.2	86.9	68.2	75.6	78.3
Repetition+	12.3	10.3	8.9	11.6	29.4	17.4	19.3	23.8	67.2	49.4	54.1	57.1	88.5	60.1	68.9	71.4
Contr. (EN)	13.8	8.3	10.7	12.3	38.6	8.3	10.3	16.4	68.1	42.7	48.9	53.8	84.2	66.1	71.2	73.7
Contr. (ES)	15.6	7.1	11.4	12.7	30.2	7.9	8.5	15.1	73.4	46.5	53.7	56.4	87.1	70.9	77.1	81.2
Ellipsis	21.0	14.6	7.6	14.4	37.4	17.6	20.9	24.7	72.8	48.3	51.9	55.9	84.0	59.8	73.4	75.6
Ellipsis+	16.2	7.8	5.4	9.2	26.8	13.4	18.3	19.5	69.2	42.1	48.2	52.7	83.6	61.6	68.1	73.2
Discourse	16.3	12.1	8.0	11.1	34.1	17.0	17.4	21.3	64.3	50.6	53.4	54.8	89.4	57.5	70.6	73.9
Discourse+	11.7	8.2	6.2	8.5	30.5	11.3	14.1	17.7	62.8	43.1	52.1	54.6	81.7	58.2	68.9	70.6
Complex	14.9	12.9	9.3	12.3	27.6	8.2	9.3	12.8	66.3	45.7	51.0	53.1	84.1	59.8	63.7	66.2
Fillers	4.5	2.3	3.1	3.6	8.6	0.6	2.6	3.1	72.5	34.5	53.6	57.7	82.4	53.1	72.5	75.1
Others	10.6	13.8	13.8	15.7	24.7	21.5	23.4	23.8	70.6	43.6	55.1	58.3	78.7	57.1	70.1	72.3
Overall	**13.8**	**10.3**	**8.7**	**11.4**	**29.1**	**13.5**	**15.8**	**18.8**	**68.4**	**45.1**	**52.5**	**55.7**	**84.2**	**60.6**	**70.0**	**72.8**

Table 3: Component-wise evaluation of the linguistically central Flex-UD components (UPOS, HEAD, and DEPREL) together with the aggregate Flex-UD score across spoken bilingual categories for four parser settings.

therefore computed as:

$$\text{Flex} - \text{UD} = \text{round}\big(\text{raw}(1 - P)\big).$$

The final goal of creating Flex-UD is not to conflate model performance by using more tolerant annotations, but to capture how structurally and linguistically reasonable the produced parse is overall. In addition to the aggregate score, Flex-UD outputs component-wise explanations, token alignments, and diagnostic notes to support qualitative analysis and model comparison.

6 Results and Discussion

6.1 Overall Parsing Performance

The combined results from Tables 2 and 3 reveal a consistent pattern across all evaluated systems and linguistic categories. Overall, the stronger systems, especially the DECAP GPT-4.1 and DECAP Llama-3-70B models, substantially outperform the weaker systems across all parsing and tagging metrics, indicating major improvements in robustness to non-canonical and discourse-rich language. However, even the best-performing systems continue to struggle with informal conversational phenomena such as fillers, contractions, ellipsis, and discourse-level structures, demonstrating that syntactic analysis of interactional language remains a difficult problem.

Across both tables, lexical and morphosyntactic recognition is consistently more robust than full syntactic parsing. UPOS tagging and Flex-UD achieve substantially higher scores than strict dependency metrics such as LAS and UAS, suggesting that the systems can often recognize word categories and recover partial local structure even when they fail to reconstruct full hierarchical dependency relations. This difference is especially visible in Table 3, where Flex-UD scores remain relatively high despite lower head and dependency attachment scores, indicating latent structural robustness not captured by strict UD evaluation alone.

These findings replicate the general tendency observed in (Kellert et al., 2025), while extending the analysis to different parser architectures and LLM families.

6.2 Human vs Parser Difficulty

The survey results shown in Figure 2 strongly support the interpretation that human annotation difficulty and parser difficulty do not always align. Human annotators rated ellipsis and highly complex constructions as the most cognitively demanding categories, with these phenomena receiving the highest scores across annotation time, ambiguity, and cognitive complexity. By contrast, fillers and contractions were rated as comparatively easier and faster to annotate despite their non-canonical spoken character. This pattern differs substantially from the parser evaluation results in Tables 2 and 3, where fillers and contractions produced some of the lowest parsing scores across systems, while several elliptical constructions were handled comparatively better by the stronger models.

This discrepancy suggests an important distinction between human linguistic processing and automatic dependency parsing. Human annotators appear to rely heavily on discourse-pragmatic expectations and interactional competence when interpreting fillers and contractions, making these phenomena relatively easy to process despite their reduced or fragmented form. Current parsing systems, however, may lack sufficient exposure to such interactional phenomena during training, since fillers and spoken disfluencies remain relatively underrepresented in standard treebanks and written corpora. Elliptical constructions, by contrast, seem to pose slightly fewer difficulties for parsers, possibly because certain analogous forms of ellipsis or omission are also attested in written language and therefore occur more frequently in training data. This partial overlap may allow models to generalize somewhat more effectively to elliptical structures than to highly interactional spoken phenomena such as fillers and contractions. The results therefore point to a broader cognitive and annotation-level mismatch between human dependency parsing strategies and the statistical learning biases of contemporary NLP systems.

6.3 Implications for UD Annotation and Evaluation.

Even though parsers show slightly greater robustness to certain elliptical constructions than to fillers, ellipsis and discourse-level phenomena continue to produce substantial degradation across systems. Our findings expose a tension between surface-oriented approaches such as Surface-Syntactic Universal Dependencies (SUD), which avoid aggressive reconstruction of omitted structure (Gerdes et al., 2019a), and inference-based approaches that posit unpronounced but structurally recoverable material (Merchant, 2001; Hardt and Romero, 2004). In spontaneous bilingual conversation, constructions that represent disfluencies such as repetition and ellipsis are frequently discourse-dependent and structurally underspecified, leading to high annotation cost and lower attachment accuracy even for strong parsers. The continued degradation observed across systems therefore suggests that the challenge lies not only in model capacity or multilingual representation, but also in the representational assumptions underlying dependency annotation itself. Spoken conversational language often contains structures that are interactionally interpretable but only partially specified syntactically, making dependency reconstruction substantially more difficult than in canonical written language.

7 Conclusion

Spoken bilingual conversational language exposes a persistent mismatch between written-language parsing assumptions and the structural variability of interactional speech. To address these challenges, we introduced a linguistically grounded taxonomy of conversational bilingual phenomena, **SpokeBench**, an expert-annotated benchmark for spoken conversational parsing, **Flex-UD**, an ambiguity-aware evaluation metric designed for structurally variable conversational language, and **DECAP**, a modular agentic parsing framework that separates spoken-phenomena handling from core syntactic analysis. Experiments across both proprietary and open-weight LLMs demonstrate substantial improvements in robustness to conversational phenomena, while Flex-UD reveals meaningful structural consistency and parser differences that remain partially hidden under strict attachment-based evaluation. Together, these findings suggest that progress in conversational bilingual parsing depends not only on stronger models but also on evaluation and annotation frameworks that better account for ambiguity, recoverability, and the discourse-driven nature of spontaneous speech.

Limitations

This work focuses on depth and linguistic complexity rather than scale. **SpokeBench** is intentionally small and curated, prioritizing expert annotation and challenging Spoken Bilingual Conversation phenomena over broad coverage across speakers, dialects, or interactional contexts. While this design enables detailed analysis, extending the benchmark to larger and more diverse spoken corpora would strengthen generalizability. In addition, our experiments focus on English-Spanish code-switching; although the identified phenomena are common in many bilingual settings, their distribution and interaction may vary across language pairs and sociolinguistic environments. Finally, we proposed an instruction-grounded framework rather than a trained parser, which enables extensibility without retraining, but is comparatively slower and depends on the quality of prompts and the underlying language models. Exploring hybrid approaches that combine spoken-aware system design with learned representations remains an important avenue for future research.

Ethical Considerations

This work uses data from the Miami Corpus of English-Spanish code-switching (Deuchar et al., 2014), which was collected and released for research purposes; no new data were collected and no personally identifiable information was accessed. While our methods aim to improve parsing and evaluation of spoken bilingual language, LLM-based annotation may propagate biases or misrepresent code-switching practices if applied without linguistic oversight, and should therefore be used cautiously outside research settings. We used AI-assisted tools (e.g., ChatGPT and Grammarly) solely for grammatical refinement and clarity of presentation.

References

Xuefeng Bai, Jialong Wu, Yulong Chen, Zhongqing Wang, Kehai Chen, Min Zhang, and Yue Zhang. 2025. Constituency parsing using llms. *IEEE Transactions on Audio, Speech and Language Processing*, 33:3762–3775. Vol. 33.

Irshad Bhat, Riyaz Ahmad Bhat, Manish Shrivastava, and Dipti Misra Sharma. 2018. Universal dependency parsing for hindi-english code-switching. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 987–998.

Özlem Çetinoğlu and Çağrı Çöltekin. 2019. Challenges of annotating a code-switching treebank. In *Proceedings of the 18th international workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 82–90.

Özlem Çetinoğlu and Çağrı Çöltekin. 2023. Two languages, one treebank: building a turkish–german code-switching treebank and its challenges. *Language Resources and Evaluation*, 57(2):545–579.

Frances Adriana Laureano De Leon, Harish Tayyar Madabushi, and Mark Lee. 2024. Code-mixed probes show how pre-trained models generalise on code-switched text. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3457–3468.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational Linguistics*, 47(2):255–308.

Margaret Deuchar, Peter Davies, Judith Herring, María C. Parafita Couto, and Dan Carter. 2014. Building bilingual corpora. In Enlli M. Thomas and Ineke Mennen, editors, *Advances in the Study of Bilingualism*, pages 93–110. Multilingual Matters, Bristol.

Kaja Dobrovoljc. 2022. Spoken language treebanks in universal dependencies: An overview. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1798–1806.

Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2019a. Improving surface-syntactic universal dependencies (sud): Mwes and deep syntactic features. In *Proceedings of the 18th international workshop on treebanks and linguistic theories (TLT, SyntaxFest 2019)*, pages 126–132.

Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2019b. Improving surface-syntactic universal dependencies (sud): surface-syntactic relations and deep syntactic features. In *TLT 2019-18th International Workshop on Treebanks and Linguistic Theories*, pages 126–132. Association for Computational Linguistics.

Carlos Gómez-Rodríguez, Muhammad Imran, David Vilares, Elena Solera, and Olga Kellert. 2024. Dancing in the syntax forest: fast, accurate and explainable sentiment analysis with salsa. In *SEPLN–CEDI-PD 2024. Seminar of the Spanish Society for Natural Language Processing: Projects and System Demonstrations*, volume 3729 of *CEUR Workshop Proceedings*, pages 12–17, A Coruña, Spain.

Daniel Hardt and Maribel Romero. 2004. Ellipsis and the structure of discourse. In *Proceedings of SALT 14*.

Muhammad Huzaifah, Weihua Zheng, Nattapol Chanpaisit, and Kui Wu. 2024. Evaluating code-switching translation with large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6381–6394.

Muhammad Imran, Olga Kellert, and Carlos Gómez-Rodríguez. 2026a. A syntax-injected approach for faster and more accurate sentiment analysis. *PeerJ Computer Science*, 12:e3519.

Muhammad Imran, Olga Zamaraeva, and Carlos Gómez-Rodríguez. 2026b. Synner: Syntax-infused named entity recognition in the biomedical domain. *JAMIA Open*, 9(1):ooaf149.

Jesin James, Vithya Yogarajan, Isabella Shields, Catherine I Watson, Peter Keegan, Keoni Mahelona, and Peter-Lucas Jones. 2022. Language models for code-switch detection of te reo māori and english in a low-resource setting. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 650–660.

Sylvain Kahane, Bernard Caron, Emmett Strickland, and Kim Gerdes. 2021. Annotation guidelines of ud and sud treebanks for spoken corpora. In *Proceedings of the 20th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2021)*, pages pp–35. Association for Computational Linguistics.

Olga Kellert, Nemika Tyagi, Muhammad Imran, Nelvin Licona-Guevara, and Carlos Gómez-Rodríguez. 2025. Parsing the switch: LLM-based universal dependency annotation for code-switched language. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 15934–15949, Suzhou, China. Association for Computational Linguistics.

Kristopher Kyle, Masaki Eguchi, Aaron Miller, and Theodore Sither. 2022. A dependency treebank of spoken second language english. In *Proceedings of the 17th workshop on innovative use of NLP for building educational applications (BEA 2022)*, pages 39–45.

Jason Merchant. 2001. *The Syntax of Silence: Sluicing, Islands, and the Theory of Ellipsis*. Oxford University Press.

Minh Van Nguyen, Viet Dac Lai, Amir Pouran Ben Veyseh, and Thien Huu Nguyen. 2021. Trankit: A light-weight transformer-based toolkit for multilingual natural language processing. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 80–90. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666. European Language Resources Association (ELRA).

Emiliana Pulido, Robert Pugh, and Zoey Liu. 2025. I speak for the árboles: Developing a dependency treebank for spanish l2 and heritage speakers. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 814–822.

Manuela Sanguinetti, Cristina Bosco, Lauren Cassidy, Özlem Çetinoğlu, Alessandra Teresa Cignarella, Teresa Lynn, Ines Rehbein, Josef Ruppenhofer, Djamé Seddah, and Amir Zeldes. 2023. Treebanking user-generated content: a ud based overview of guidelines, corpora and unified recommendations. *Language Resources and Evaluation*, 57(2):493–544.

Igor Sterner and Simone Teufel. 2025a. Code-switching and syntax: A large-scale experiment. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11526–11533, Vienna, Austria. Association for Computational Linguistics.

Igor Sterner and Simone Teufel. 2025b. Minimal pair-based evaluation of code-switching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18575–18598, Vienna, Austria. Association for Computational Linguistics.

Yuanhe Tian, Fei Xia, and Yan Song. 2024. Large language models are no longer shallow parsers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7131–7142.

Ziyan Zhang, Yang Hou, Chen Gong, and Zhenghua Li. 2025. Self-correction makes LLMs better parsers. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 6749–6762, Suzhou, China. Association for Computational Linguistics.

A Distribution of SpokeBench

Table 4 reports the distribution of sentences in **SpokeBench** by Spoken Bilingual Conversation phenomenon. The benchmark was designed to balance coverage of frequent spoken phenomena with a controlled set of highly complex constructions, enabling systematic evaluation of parsers under varied spoken-language conditions.

Category	# Sentences
Repetition	10
Repetition+	20
Contractions English (EN)	10
Contractions Spanish (ES)	10
Ellipsis	10
Ellipsis+	20
Discourse	10
Discourse+	20
Fillers	10
Complex (3+ phenomena)	20
Others	20
Total	**160**

Table 4: Composition of the SpokeBench benchmark by Spoken Bilingual Conversation phenomenon.

B Examples of Spoken Bilingual Conversation Phenomena

Table 5 provides representative examples of the spoken code-switching phenomena included in our taxonomy, drawn from the Miami Corpus. Each example illustrates a characteristic non-canonical structure encountered in conversational bilingual speech, along with an English translation to support interpretability for non-Spanish speakers.

C Condensed Annotation Guidelines

C.1 General Annotation Principles

Annotations in SpokeBench follow UD conventions with targeted extensions for spoken, disfluent, and code-switched data, guided by the principles below:

- **UD-first consistency:** Annotators adhered to standard UD conventions unless a spoken-language phenomenon required an explicit, documented deviation.
- **Single-root constraint:** Every sentence was required to contain exactly one syntactic root, including minimal, elliptical, or fragmentary utterances (e.g., *si mmh*). The root was assigned to the lexical item carrying the greatest semantic weight.
- **Ambiguity tolerance:** Structural ambiguity was treated as an inherent property of spoken conversational bilingual data rather than as an annotation error. Annotators were instructed to prefer linguistically plausible analyses over maximally specific ones.
- **Internal consistency:** All annotations were reviewed for internal consistency, with particular attention to agreement between UPOS tags and dependency relations.
- **Conservative tokenization:** Original corpus tokenization was preserved whenever possible. Token splitting or insertion of additional rows was permitted only when necessary to support accurate syntactic analysis.

C.2 Treatment of Spoken-Language Phenomena

This subsection summarizes the core annotation rules applied to recurring spoken-language phenomena in SpokeBench. These rules were designed to minimize speculative reconstruction while ensuring consistent and transparent treatment across annotators.

Repetition and Repairs Repeated tokens that function as speech repairs or disfluencies were annotated using the dependency relation `reparandum`. Annotators were instructed to identify the intended syntactic structure and attach repeated or overridden material as dependents of the corresponding head in the corrected structure. This procedure was applied uniformly to both monolingual and code-switched repetitions.

Contractions English and Spanish contractions (e.g., *wasn't*, *del*) were split into their component morphemes by inserting additional tokens immediately following the original form. The expanded tokens were then annotated according to standard UD conventions. This approach preserves morphological transparency while enabling accurate dependency assignment.

Multiword Expressions and Compounds Fixed expressions and conventionalized compounds (e.g., *you know*, *a lot*) were treated as single syntactic units rather than annotated compositionally. Annotators combined such expressions using underscore-separated tokens and assigned UPOS and dependency relations corresponding to their discourse or syntactic function.

Category	Example (Miami Corpus)	English Translation
Repetition	entonces then I will I will I will not be buying any stuff for you this weekend .	So then I will, I will, I will not be buying any stuff for you this weekend.
Discourse Elements	uhhuh bueno es verdad .	Uh-huh, well, it's true.
Ellipsis	pero si they are covered from .	But if they are covered from [it].
Contractions	and tú sabes it wasn't the same .	And you know it wasn't the same.
Compound Words	cuando uno va al swimming pool ahí no me gusta .	When one goes to the swimming pool, I don't like it there.
False Start	you have you gotta show tu tía .	You have.. you gotta show your aunt.
Filler Words	fíjate que they gave them an honorary uh diploma .	Look, they gave them an honorary, uh, diploma.
Slang / Curse Words	I mean I'm thinking coño if I'm not here what the hell would happen to me ?	I mean, I'm thinking, damn, if I'm not here, what the hell would happen to me?
Enclisis	more jugo dile a Carla que te dé more jugo	More juice, tell Carla to give you more juice.

Table 5: Representative examples of some spoken bilingual conversational language phenomena from the Miami Corpus, with English translations.

Ellipsis and Fragmented Utterances Elliptical or truncated constructions were handled according to the recoverability of the intended meaning. Tokens that could be removed without altering the core interpretation of the utterance were labeled as `reparandum`. Tokens whose syntactic heads were missing and could not be confidently reconstructed were assigned the relation `dep` and attached to the sentence root. In fragmentary utterances lacking an explicit predicate, the most semantically salient token was designated as the root.

Discourse Markers and Fillers Discourse markers, interjections, and hesitation sounds (e.g., *uh*, *mmh*) were annotated with `UPOS=INTJ` and assigned the dependency relation `discourse`. Fillers were treated as a distinct subclass to facilitate later analysis but followed the same structural annotation principles.

C.3 Review and Adjudication Procedure

Each sentence in SpokeBench was independently annotated by at least two annotators. Annotations were subsequently reviewed using a graded acceptability scale that distinguished between acceptable variation and substantive errors. Reviewers were instructed to assess whether an analysis was linguistically defensible under UD rather than whether it matched their personal preference. Sentences flagged as borderline or unacceptable were escalated for expert discussion. Final annotations were determined through deliberation involving senior linguists on the project, ensuring consistency across the benchmark. This process resulted in a gold-standard dataset that reflects informed consensus while acknowledging the inherent ambiguity of spoken bilingual conversations.

D DECAP Interaction Protocol

The details of the DECAP agent workflow are shown below, in Algorithm 1.

Algorithm 1 DECAP per-sentence interaction rule

Require: Tokenized sentence $S = \{(t_i, \text{lang}_i)\}_{i=1}^{n}$
Ensure: UD parse with validated structure and confidence annotations
1: $\mathcal{S} \leftarrow \text{SPH}(S)$ {Detect spoken phenomena; propose tokenization edits}
2: $\mathcal{L} \leftarrow \text{LSR}(\mathcal{S})$ {Apply language-specific normalization and MWEs}
3: $\mathcal{C} \leftarrow \textsc{Core}(\mathcal{L})$ {Assign UPOS, heads, and dependency relations}
4: $\mathcal{F} \leftarrow \text{V/R}(\mathcal{S}, \mathcal{L}, \mathcal{C})$ {Enforce single root, acyclicity, and consistency}
5: $\mathcal{F}$ includes confidence scores and logs any structural repair
6: **return** $\mathcal{F}$

E FLEX-UD Severity Examples

Catastrophic Errors. Catastrophic errors correspond to violations that undermine the global structural validity or interpretability of a parse. These include cases where a dotted multiword expression (MWE) is required by the gold annotation but omitted by the system; reparandum tokens attached to unrelated subtrees, resulting in incorrect root selection or major structural distortion; and invalid head references that persist after canonicalization. Such errors receive large penalty contributions in Flex-UD, typically in the range of $P = 0.25$ to 0.6 per issue (clipped), reflecting their disproportionate impact on parse quality.

Minor Errors. Minor errors correspond to linguistically plausible deviations that do not compromise the overall structure of the parse. Examples include tolerant POS substitutions (e.g., VERB↔AUX), near-miss dependency relations (e.g., `obj`↔`obl`), and other UPOS or DEPREL mismatches that fall within predefined closeness classes. These errors incur small penalty contributions, typically in the range of $P = 0.01$ to 0.05 per issue, allowing Flex-UD to distinguish acceptable variation from substantive failure.

F Prompts Used for DECAP

The prompt structures for the 4 agents used in the DECAP pipeline are described below. Each prompt corresponds to a dedicated agent with a narrowly scoped responsibility in the annotation process, progressing from spoken-phenomena detection to language-specific normalization, core UD assignment, and final verification and ranking.

Prompt for Spoken-Phenomena Handler (SPH)

System Prompt (SPH).
You are the **Spoken-Phenomena Handler (SPH)** for Spanish–English conversational data. Your responsibility is to detect spoken-language phenomena, propose minimal tokenization edits (e.g., contraction splits and multiword expressions), and produce a single structured JSON object for downstream agents. Return *only* valid JSON and behave deterministically.

User Prompt (SPH).
Input: A JSON array of tokens representing one sentence, including sentence ID, token index, surface form, and language tag.
Goal: Produce a validated SPH JSON object that:

- identifies spoken-language phenomena (e.g., repetition, ellipsis, discourse, fillers),
- proposes necessary tokenization edits (contractions, enclisis, MWEs),
- constructs a complete and consistent proposed ID mapping, and
- reports a brief summary and confidence score.

Key Rules (excerpt):

- Contractions and enclitic forms are split only when they correspond to multiple UD nodes.
- Fixed multiword expressions are collapsed into dotted nodes when they function as a single syntactic unit.
- Repetitions and repairs are marked as `reparandum` and linked to the intended head.
- Elliptical or stranded tokens are labeled conservatively and anchored to the root when necessary.

Additional rules governing ID assignment, mandatory MWEs, validation checks, and output schema are provided in the full prompt specification in the GitHub (. . .).

Output: A single JSON object containing normalized tokens, spoken-phenomena labels, a proposed ID map, confidence scores, and brief summary notes. No explanatory text is permitted.

Prompt for Language-Specific Resolver (LSR)

System Prompt (LSR).
You are the **Language-Specific Resolver (LSR)** for Spanish–English code-switched conversational data. Your single responsibility is to accept the SPH JSON and apply conservative language-specific normalization (contraction/enclisis expansion, MWE confirmation/creation, lemma suggestions, and language-confidence scores). Respect SPH edits; return *only* one JSON object and behave deterministically.

Input: The SPH JSON object with keys including `sentence_id`, `original_tokens`, `tokens`, and `proposed_id_map`.
Goal: Produce an updated JSON that:

- confirms or conservatively expands standard contractions/enclitics (INTEGER-SHIFT),
- confirms or (rarely) creates high-precision MWEs as dotted nodes (whitelist + conservative rule),
- supplies lowercase `lemma` suggestions and `lsr_confidence` per token,
- updates `proposed_id_map` consistently for any insertions.

Key Rules (excerpt):

- **Respect SPH authority**: do not undo SPH tokenization edits; only add splits when SPH omitted a clear, standard contraction/enclitic.
- **Contraction handling (INTEGER-SHIFT)**: expand only standard contractions that map to separate UD nodes; insert new integer proposed_IDs and shift subsequent IDs +1, reflecting changes in `proposed_id_map`.
- **MWE creation (DOTTED IDs)**: auto-combine mandatory whitelist MWEs (case-insensitive). Create other MWEs only under conservative, high-confidence criteria; add dotted node `<start>.1` placed after the span.
- **Lemmas & language confidence**: provide best-effort lemmas (verbs → infinitive, nouns → singular) or `null`; set `lsr_confidence` in [0.0,1.0] and one-line `lsr_notes` for nontrivial edits.
- **IDs and validation**: do not change original `token_index`; ensure unique proposed_IDs, complete `proposed_id_map`, and consistency after integer-shifts.

Examples (excerpt):

- **Contraction split**: `"don't"` → `"do"` (orig id) + inserted `"not"` (new id); update map so orig id maps to `["5","6"]`.
- **MWE (whitelist)**: `"pitta bread"` → keep 6,7 integer rows and add dotted 6.1 with `split_token="pitta_bread"`; reflect in map.

Additional details on whitelist entries, edge cases, full schema, and validations are available in the full prompt spec on the project repository (. . .).

Output: A single JSON object with keys `"sentence_id"`, `"tokens"` (including `proposed_ID`, `split_token`, `lang_tag`, `lemma`, `lsr_notes`, `lsr_confidence`, `mwe`), `"proposed_id_map"`, and `"summary_notes"`. No additional text is permitted.

Prompt for Core UD Assigner (Core)

System Prompt (Core).
You are the **Core UD Assigner** for Spanish–English conversational data. Your task is to assign UD-style annotations (UPOS, HEAD_ID, DEPREL) following the Miami Gold Subset spoken-language rules. Respect tokenization and proposed_IDs from upstream (SPH + LSR) and spoken-language directives (spoken_label / spoken_anchor). Annotate only dotted MWE nodes (n.1); leave MWE component integer rows unannotated. Ensure exactly one root. Return *only* a single JSON object (no commentary).

User Prompt (Core).
Input: LSR JSON object with keys: `sentence_id`, `tokens` (`proposed_ID`, `split_token`, `lang_tag`, `lemma`, `lsr_confidence`, `mwe`, `spoken_label`, `spoken_anchor`), and `proposed_id_map`.
Goal: Produce an annotated JSON with UD fields for every annotatable node, obeying spoken-label mappings and UD constraints.

Key Principles (high-level):

- **Respect upstream**: follow SPH/LSR spoken_label and recommended head (spoken_anchor) when provided.
- **Spoken-label → UD mapping**: e.g., `reparandum` → DEPREL=`rep`; `dep` → DEPREL=`dep`; `discourse/filler` → UPOS=`INTJ` and DEPREL=`discourse` (attach to root unless syntactically integrated).
- **Single root**: exactly one token with `HEAD_ID="0"` and `DEPREL="root"`; prefer finite VERB, else central NOUN/PRON, else communicative token.
- **Allowed tag sets**: UPOS and DEPREL must be chosen from the Miami-approved lists (use only allowed values).

Output Schema (required, single JSON object):

```
{
 "sentence_id": "<id>",
 "annotated_tokens": [
   {
     "proposed_ID": "<str>",
     "FORM": "<split_token or empty>",
     "LEMMA": "<lemma or empty>",
     "UPOS": "<allowed UPOS or empty>",
     "HEAD_ID": "<proposed_ID or '0' or empty>",
     "HEAD_FORM": "<FORM of head or 'root' or empty>",
     "DEPREL": "<allowed DEPREL or empty>",
     "core_confidence": 0.0-1.0,
     "core_notes": "<one-line justification>"
   }, ...
 ],
 "summary_notes": "<one-line summary>"
}
```

Essential validations (must pass): ...

Return: exactly one JSON object following the schema above. Do not include any additional text.

Prompt for Verifier & Ranker (V/R)

System Prompt (V/R).
You are the Verifier & Ranker. Merge SPH, LSR and Core outputs for one sentence; enforce ID/HEAD consistency; repair structural issues (single root, cycles); compute per-token `final_confidence` and `penalty`; remap proposed_IDs to sheet integers when requested; and emit sheet-ready rows plus an adjudication log. Respect authoritative ordering: Core ▷ LSR ▷ SPH. Return *only* the JSON described below. **User Prompt (v/R).**

Input & Goal.
Input: JSON bundle with keys `"sph"`, `"lsr"`, `"core"`. Goal: produce validated final token table (all integer rows, inserted integers, dotted MWE nodes), mapping to sheet IDs, plus `adjudication_log` and a one-line `final_summary`.

Condensed procedure (deterministic):

- Canonicalize nodes from SPH original order, inserting LSR/SPH-added integers and dotted MWEs.
- Merge annotations (prefer Core; fill from LSR then SPH).
- Validate/remap HEADs (HEAD_FORM match, then SPH spoken_anchor, else attach to root) and log.
- Enforce single root by priority (finite VERB, AUX, central NOUN/PRON, else highest confidence) and log.
- Detect repair cycles using combined_conf; reattach lowest node(s) to root until acyclic; log numeric rationale.
- Compute final_confidence and penalty (per rules) and add adjudication notes.
- Remap to sheet_IDs (1..N) and produce sheet_HEAD_ID mapping.
- Emit final JSON and validate

Required output shape (exact):

```
{
 "sentence_id":"<id>",
 "final_tokens":[
   {
     "sentence_id":<id>,
     "orig_token_index":<int>,
     "split_token":"<str>",
     "ID":"<proposed_ID>",
     "sheet_ID":<int>,
     "FORM":"<str or blank>",
     "LEMMA":"<str or blank>",
     "UPOS":"<or blank>",
     "HEAD_ID":"<proposed_ID or '0'>",
     "sheet_HEAD_ID":<int or 0>,
     "HEAD":"<HEAD_FORM or blank>",
     "DEPREL":"<or blank>",
     "final_confidence":0.0-1.0,
     "penalty":0.0-1.0,
     "adjudication_note":"<short>"
   },
   ...
 ],
 "adjudication_log":[ "...", "..." ],
 "final_summary":"<one-line>"
}
```

Determinism & logging: use deterministic tie-breakers (prefer lower sheet_ID). Each structural fix must add one bullet to `adjudication_log` stating what changed, why (include numeric confidences), and which signals supported the decision.