# NM at CRF Filling 2026: A Two-Stage LLM Pipeline for Clinical CRF Population

**Niccolò Morabito**
morabito.niccolo@gmail.com

## Abstract

This paper describes our participation in the CRF Filling Shared Task 2026, which aims to automatically populate a predefined Case Report Form (CRF) from clinical notes describing patients with dyspnea. We propose a two-stage pipeline based on large language models (LLMs). In the first stage, a few-shot prompted LLM extracts candidate CRF fields from the clinical note and outputs them in a structured JSON format. In the second stage, a separate LLM verifies each extracted field against the original note and removes predictions that are not supported by explicit textual evidence. This verification step aims to reduce false positives generated during extraction. Experiments on the development set show that the verification stage significantly reduces unsupported predictions while preserving most correct extractions, resulting in improved macro F1. On the official test set, the proposed system achieves a macro F1 score of 0.56 for both English and Italian. These results indicate that separating extraction and verification can balance recall-oriented extraction with precision-oriented validation in CRF population tasks.

## 1 Introduction

Automatically converting unstructured clinical narratives into structured data is an important step for enabling large-scale clinical research and clinical decision support systems. In many medical workflows, relevant patient information is recorded in free-text notes, making large-scale analysis difficult without structured representations.

In this context, the CRF Filling Shared Task (Ferrazzi et al., 2026b) focuses on populating a predefined Case Report Form (CRF) from clinical notes describing patients presenting with dyspnea.

To address this problem, we propose a two-stage pipeline combining extraction with conservative verification. In the first stage, a few-shot large language model (LLM) extracts candidate CRF fields directly from the clinical note. In the second stage, a separate LLM verifies each extracted field against the original note, removing predictions that are not supported by explicit textual evidence. This design aims to balance coverage and precision by combining the generative capabilities of LLMs with a verification step that reduces unsupported predictions.

## 2 Task Description

The shared task consists of automatically populating a CRF from free-text clinical notes describing patients presenting with dyspnea. The CRF contains a fixed set of 134 clinical items, each corresponding to a specific variable such as symptoms, clinical findings, or laboratory measurements. For each item, systems must select one value from a predefined list of valid options based solely on the information contained in the clinical note. The same CRF structure is used for all notes.

A key challenge of the task is the extreme sparsity of annotations. In many cases, the clinical note does not contain information relevant to most CRF items. When an item cannot be populated from the note, it is assigned the value *unknown*. As a result, approximately 95% of the CRF fields are labeled as *unknown* across the dataset, requiring systems to extract information only when explicit textual evidence is present.

The training data combines multiple sources of supervision, including a small set of 10 gold-standard pairs of clinical notes and filled Dyspnea CRFs (Kaczmarek et al., 2026), a collection of semi-automatically annotated CRFs for other medical conditions (Ferrazzi et al., 2025), and a set of 2,667 unannotated clinical notes describing patients with dyspnea (Ferrazzi et al., 2026a). In our experiments, we only use the 10 gold-standard pairs as few-shot examples for prompting the extraction model (see 4.1).



*Proceedings of the BioNLP 2026 (Shared Tasks)*, pages 1–5

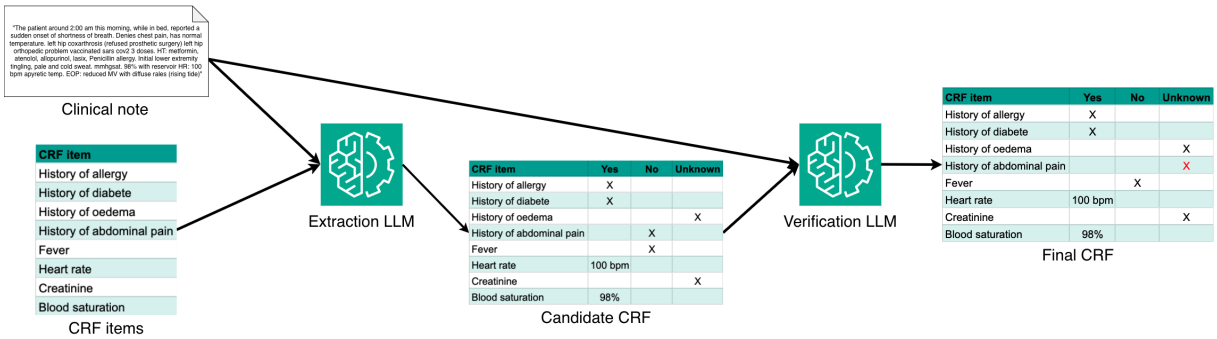


Figure 1: Overview of the two-stage pipeline for CRF extraction and verification. The first stage extracts candidate CRF fields from clinical notes using a few-shot LLM, and the second stage verifies each field against the original note to remove unsupported predictions to generate the final CRF.

| System | Model | TP | FP | FN | F1 |
|---|---|---|---|---|---|
| Extraction | Gemini 3 Flash | 278 | 420 | 83 | 0.584 |
| Extraction | LLaMA4 Maverick 17B | 179 | 131 | 170 | 0.578 |
| Extraction + Verification | Gemini 3 Flash + LLaMA4 Maverick 17B | 274 | 373 | 88 | 0.612 |

Table 1: Performance comparison on the English development set.

For evaluation, the organizers provide a development set of 80 gold-standard note-CRF pairs and a test set of 200 gold-standard pairs (Kaczmarek et al., 2026). The task is released in two languages, English and Italian, with parallel datasets available for both languages.

## 3 System Overview

Our system follows a two-stage pipeline designed to balance recall and precision when extracting structured information from clinical notes. The first stage performs CRF extraction using a few-shot prompted LLM. Given a clinical note, the model generates a candidate CRF containing only the CRF fields that are explicitly supported by the text (see Figure 1).

Preliminary experiments showed that this extraction stage tends to over-predict fields, producing a relatively high number of false positives. To mitigate this issue, we introduce a second verification stage. In this step, a different LLM receives both the original clinical note and the CRF generated in the first stage and verifies whether each predicted field is supported by explicit textual evidence. Fields that cannot be directly justified by the text are removed from the output of the final CRF.

This two-stage design allows the system to combine the strong extraction capabilities of large generative models with a more conservative verification mechanism aimed at reducing unsupported predictions.

## 4 Method

This section describes the two stages of the proposed pipeline in detail.

### 4.1 CRF Extraction with Few-Shot LLM

The extraction stage formulates CRF completion as a structured information extraction problem solved through few-shot prompting of a large language model. Few-shot learning allows a model to adapt to a new task by conditioning on a small number of input–output examples provided directly in the prompt.

The shared task provides a very small set of gold-standard annotated examples (10 note–CRF pairs). These examples can be used as demonstrations in the prompt to guide the model toward the desired behavior. In our experiments, we include three examples in the prompt, which provided a good balance between task guidance and prompt length.

Given a clinical note, the model receives a system prompt describing the extraction task and specifying strict constraints on how CRF fields should be produced. The model is prompted once per clinical note to generate a structured JSON object containing all the CRF fields that can be populated from the text, rather than generating each field independently. In particular, the prompt instructs the model to extract only fields that are supported by

| System | Model | TP | FP | FN | F1 |
|---|---|---|---|---|---|
| Extraction | Gemini 3 Flash | 289 | 385 | 56 | 0.599 |
| Extraction | GPT-OSS 120B | 231 | 190 | 121 | 0.575 |
| Extraction + Verification | Gemini 3 Flash + GPT-OSS 120B | 276 | 372 | 78 | 0.604 |

Table 2: Performance comparison on the Italian development set.

clear and explicit evidence in the clinical note and to avoid any form of inference or interpretation beyond the literal text. The full prompt used in the extraction stage is reported in Appendix A.1.

The 3-shot examples included in the prompt consist of pairs of clinical notes and their corresponding CRF representations, demonstrating the expected input–output format.

For this stage we use the Gemini 3 Flash model (Pichai, 2025), which showed the strongest extraction performance among the models evaluated during preliminary experiments. These experiments consisted of running different LLMs under the same prompting setup on the development set and comparing their macro F1 scores. In addition to Gemini 3 Flash, we tested open-weights models such as LLaMA4 Maverick 17B, Gemma3 27B, Qwen3, and GPT-OSS 120B, which achieved lower extraction performance when used alone. For conciseness, these additional results are not reported in detail in this paper, as they do not affect the main findings.

### 4.2 CRF Verification

Although the extraction stage is explicitly instructed to output only fields supported by clear textual evidence, we observed that the model often produces unsupported predictions, resulting in a relatively high number of false positives (particularly with the Gemini 3 Flash model).

To address this issue, we introduce a second verification stage. In this step, a separate LLM receives two inputs: the original clinical note and the CRF generated by the extraction stage. The task of this model is to verify whether each predicted field is directly supported by the text.

The verification prompt instructs the model to examine each populated CRF field and retain it only if the clinical note contains clear and unambiguous evidence supporting the assigned value. If the evidence is absent, ambiguous, or implicit, the field must be removed from the output. Importantly, the model is explicitly prohibited from adding new fields or modifying the values of existing predictions. Its role is strictly limited to filtering unsupported predictions. The complete prompt used in this stage is provided in Appendix A.2.

For this stage we selected, among the models evaluated in the preliminary experiments mentioned in Section 4.1, those that exhibited the most conservative prediction behavior, i.e., producing fewer false positives and fewer overall predictions when used for extraction. This selection was performed separately for each language based on development set performance. In particular, we used the LLaMA4 Maverick 17B model for the English language and GPT-OSS 120B for Italian (OpenAI et al., 2025), as these models showed the strongest tendency to refrain from assigning CRF labels in the absence of clear textual evidence in their respective settings.

The final CRF prediction corresponds to the filtered JSON output produced by this verification stage.

## 5 Results

We evaluate our approach on both the development and test sets provided by the shared task. Performance is measured using macro F1, the official evaluation metric of the competition. The development set allows a more detailed analysis of system behavior, including the number of true positives (TP), false positives (FP), and false negatives (FN), while for the test set only the final macro F1 score is available.

Tables 1 and 2 report the results on the development set for the English and Italian datasets, respectively. We compare the extraction performance of two models used independently with the final two-stage pipeline.

The Gemini 3 Flash model achieves the best extraction performance (also compared to other models tested in preliminary experiments, which are not shown here for brevity). However, it produces a relatively high number of false positives in both English and Italian.

In contrast, on the English dataset (Table 1), LLaMA4 Maverick 17B produces fewer false pos-

itives but suffers from a higher number of false negatives due to its more conservative prediction behavior. The proposed pipeline combines these complementary characteristics by using Gemini for the extraction step and LLaMA4 for the verification step. This second stage filters unsupported predictions while preserving most of the correct extractions. As a result, the verification stage significantly reduces the number of false positives produced by the extraction model without substantially decreasing the number of true positives, leading to a noticeable improvement in macro F1.

For the Italian dataset, a different model was used in the verification stage. Preliminary experiments showed that GPT-OSS 120B achieved slightly better performance in the extraction+validation pipeline. More generally, we observed a similar behavior between LLaMA4 Maverick 17B and GPT-OSS models: compared to the models used for extraction, they tend to produce fewer predictions and adopt a more conservative strategy. While this behavior may reduce recall when used alone, it proves beneficial in the verification step, where the goal is to filter unsupported predictions produced during extraction (Table 2).

Table 3 reports the final performance of the pipeline on the test set for both languages. As expected, the overall macro F1 score is lower than the one observed on the development set. This difference is common, as the test set represents unseen data and may contain different distributions of CRF mentions and textual patterns.

Despite this decrease, the results confirm the effectiveness of the proposed two-stage approach. The pipeline maintains competitive performance across both languages while benefiting from the verification stage. This suggests that separating extraction and validation allows the system to balance recall-oriented extraction with precision-oriented verification, leading to more reliable CRF predictions overall.

| System | Eng F1 | Ita F1 |
|---|---|---|
| Extraction + Verification | 0.56 | 0.56 |

Table 3: Final macro F1 scores of the proposed system on the test set.

## 6 Conclusion

In this paper we presented a two-stage LLM-based pipeline for automatically populating clinical Case Report Forms from free-text clinical notes. The system separates the extraction of candidate CRF fields from their verification, allowing different models to specialize in complementary behaviors. The extraction stage focuses on maximizing coverage of potentially relevant fields, while the verification stage adopts a more conservative strategy to filter predictions that are not supported by explicit textual evidence.

Experimental results show that this design helps reduce the number of false positives produced by the extraction model while preserving most correct predictions, leading to improved macro F1 on the development set. The final system achieves a macro F1 score of 0.56 on both English and Italian test sets.

Overall, our results suggest that combining generative extraction with explicit verification is a promising direction for structured information extraction from clinical narratives.

## References


Pietro Ferrazzi, Mattia Franzin, Alberto Lavelli, and Bernardo Magnini. 2026a. Small LLMs for Medical NLP: a Systematic Analysis of Few-Shot, Constraint Decoding, Fine-Tuning and Continual Pre-Training in Italian. *Preprint*, arXiv:2602.17475. ArXiv preprint.

Pietro Ferrazzi, Soumitra Ghosh, Alberto Lavelli, and Bernardo Magnini. 2026b. Overview of the CRF 2026 Shared Task on Clinical Case Report Forms filling. In *Proceedings of the Third Workshop on Patient-Oriented Language Processing (CL4Health)*, Palma, Mallorca (Spain). ELRA.

Pietro Ferrazzi, Alberto Lavelli, and Bernardo Magnini. 2025. Converting Annotated Clinical Cases into Structured Case Report Forms. In *Proceedings of the 24th Workshop on Biomedical Language Processing*, pages 307–318, Vienna, Austria. Association for Computational Linguistics.

Gabriela Anna Kaczmarek, Pietro Ferrazzi, Lorenzo Porta, Vicky Rubini, and Bernardo Magnini. 2026. Toward automatic filling of case report forms: A case study on data from an italian emergency department. *Preprint*, arXiv:2602.23062. ArXiv preprint.

OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, and 107 others. 2025. gpt-oss-120b & gpt-oss-20b Model Card. *Preprint*, arXiv:2508.10925.

Sundar Pichai. 2025. A new era of intelligence with gemini 3.

# A Appendix: Prompt Details

## A.1 Extraction Stage Prompt

The system prompt used for the CRF extraction stage is shown below:

```
You are a medical information extraction system.
     Task:

From a single clinical note, select ONLY the
    CRF fields that are supported by clear,
    explicit, and unambiguous evidence in the
    text, and assign them a value.

Critical rules:
- If a field is not EXPLICITLY and CLEARLY
    stated, DO NOT output that field.
- Absence of information, negations, hypotheses
    , suspicions, plans, history without
    confirmation, vague wording or doubt must
    result in the field being completely
    omitted.
- Only output a field when the text contains a
    direct statement that uniquely determines
    its value, and choose the value that
    exactly matches the literal meaning of the
     text.
- Never infer, guess, normalize, or interpret
    beyond the literal text.

Output rules:
- Output a valid, minified JSON object
    containing ONLY the selected fields.
- Each output value must be exactly one of the
    allowed values for that field.
- All values are strings.
- Do not include explanations, comments, or
    extra text.
```

## A.2 Verification Stage Prompt

The system prompt used for the verification stage is shown below:

```
You are a clinical data verification system.
    Task:
Given
1) a clinical note
2) a JSON object containing ONLY populated CRF
    fields
Your job is to VERIFY each field in the JSON
    against the clinical note.

Rules:
- For EACH field in the input JSON, check
    whether the clinical note contains clear,
    explicit, and direct evidence supporting
    the assigned value.
- If the evidence is sufficient and unambiguous
    , KEEP the field exactly as is.
- If evidence is absent, weak, implicit,
    ambiguous, contradictory, or based on
    inference, REMOVE the field entirely from
    the output.
- Do NOT add new fields.
- Do NOT change values.
- Do NOT normalize or reinterpret.
- Do NOT infer from medical knowledge.
- Judge ONLY based on what is explicitly
    written in the note.

Output:
- Return ONLY a valid, minified JSON object.
- Include ONLY the fields that are fully
    supported.
- No explanations, comments, or extra text.
```