LotusOrchid at #SMM4H–HeaRD 2026: Fitting pretrained encoders for Dutch medical data

Sophie Arnoult
Vrije Universiteit Amsterdam
s.i.arnoult@vu.nl

Shutao Chen
Vrije Universiteit Amsterdam
realsusana.c@gmail.com

Piek Vossen
Vrije Universiteit Amsterdam
piek.vossen@vu.nl

Abstract

This paper presents our submission to MultiClinAI's NER subtask for #SMM4H-HeaRD 2026. We focus on the questions 1) which Language Model represents the clinical notes best and 2) which annotations can help training these models. To get answers for these questions, we follow a token-based classification approach with pretrained encoder language models, where we compare models that were pretrained on generic data against medical data, and on a single language, Dutch, against many languages. In addition, we present two data-augmented systems: one with data from the other languages of the workshop for multilingual training, and one with synthetic annotations[1].

1 Introduction

The MultiClinAI's NER subtask (Gallego-Donoso et al., 2026) for #SMM4H-HeaRD 2026 (Lopez-Garcia et al., 2026) addresses the detection of *diseases*, *procedures*, and *symptoms* in clinical notes. The training and test data for this task originate from Spanish clinical texts and annotations. The Spanish texts have been translated into other languages, among them Dutch, and the annotations have been projected from Spanish to these other languages. The data have been divided into training and test data by the organizers. Our submissions focus on the Dutch test texts and annotations. The questions that we try to address are: 1) which language model represents the Dutch text best and 2) what annotations contribute most to finetune these models for the task.

We use pretrained encoders for the task (Devlin et al., 2019). Although generative approaches have been proposed for NER (Li et al., 2020; Wang et al., 2025), this allows us to compare different available and relatively fast and sustainable models for their ability to represent Dutch clinical notes:

- medRoBERTa.nl built from 10M Dutch clinical notes (Verkijk and Vossen, 2025).
- RobBERT built from Dutch Wikipedia and Web texts (Delobelle et al., 2020)
- XLM-RoBERTa: a multilingual model built from Wikipedia and Common Crawl text from a hundred languages (Conneau et al., 2019)

medRoBERTa.nl is trained to represent Dutch clinical notes and has shown good performance on other classification tasks in that domain (Chen and Vossen, 2026). While this makes it better candidate than RobBERT, it remains to be seen how these notes differ from the provided training data, not just because the Dutch training data are translated from Spanish, but also because of inherent textual differences: Verkijk and Vossen (2025) report that the language in the notes is idiosyncratic, with short and incomplete sentences, whereas the language in the training data appears to be fluent. We were curious as to how this difference would affect performance. When are specialist models too specialized and how does this show?

XLM-RoBERTa is not specifically pretrained for medical notes, but represents a wide variety of languages, which allows for transferring annotations on other languages to Dutch (Wu and Dredze, 2019). Through our submissions based on these different models, we investigate to what extent the choice of model is a factor on representing the Dutch test texts and likewise, on the Dutch task. When it comes to annotations, we can only use the projected Dutch annotations to finetune medRoBERTa.nl and RobBERT. Using XLM-RoBERTa allows us to compare the use of Dutch annotations against all annotations in a multilingual model.

[1]Code and synthetic annotations are available a https://github.com/cltl/MultiClinNER-2026

Proceedings of the 11th Social Media Mining for Health Research and Applications (SMM4H-HeaRD 2026) Workshop and Shared Tasks, pages 139–145

LLMs have proven useful for annotating data, showing both consistency and the ability to tackle specialized domains. Hsu and Roberts (2025) notably demonstrated that leveraging large language models for "knowledge-free weak supervision" could produce classifiers that outperform purely supervised baselines when gold standard data are scarce. Similarly, Oliveira et al. (2025) showed that combining prompt based language models with weak supervision provides a robust alternative to exhaustive manual annotation, noting that generated labels serve as a "valid option for model training when human annotation is expensive". Given the large scale nature of the task and annotations, could one benefit from additional LLM synthetic annotations? We tested this idea on the medical medRoBERTa.nl model, merging training and synthetic annotations at the sequence level.

2 NER as token-based classification

All our systems use transformer encoders with a classification head, trained to minimize cross-entropy loss for subtoken multi-class predictions.

2.1 Data selection

The training data for the task consist of medical texts that were automatically translated from Spanish to 6 other languages and then separately annotated for the three categories *of disease*, *procedure*, and *symptom*. As annotators were allowed to correct the underlying texts for each category, the texts do not always align across categories.

As one of the strengths of pretrained encoder models is that a single model, equipped with a classification layer, suffices to predict different categories, we were keen to align texts again for training. In the interest of time, we aligned document sequences (paragraphs or chunks[2] thereof), filtering those with matching texts; a thorough approach would have involved merging divergent sequences, correcting entity offsets where needed.

Entities of different types are merged in the resulting sequences; overlapping entity spans are filtered, keeping the entity starting first or the one with the shorter span if two entities start together. Entity merging is applied in the same way for the `nl+synth` data set that combines the workshop's training data with synthetic annotations. As Table 1 shows, sequence alignment and type merging preserve about 75% of the input Dutch entities, and 62% of the input entities across all languages. The synthetic and input annotations overlap largely, resulting in an increase of only 11% entities for the `nl+synth` dataset compared to the `nl` dataset.

	input	merged data		
	entities (k)	docs	chunks (k)	entities (k)
nl	80.9	1198	16.1	61.1
nl+synth	– +62.7	1198	16.1	69.4
all-lang	566	1258	102	352

Table 1: Training data counts before and after sequence alignment and type merging. Document counts refer to unique document ids, regardless of language or entity type. Entity counts encompass all entity types.

2.2 Encoder models as token classifiers

Input annotations are mapped to BIO tags without pretokenizing: character offsets are used to identify subtokens in entity spans, where the first subtoken is mapped to the B tag of the relevant entity type, and following subtokens to the corresponding I tag.

For inference, the subtoken level predictions are converted to entity spans by looking for B tags as the start of an entity phrase, and for the *last* I tag of the same type before the following B tag. This, for instance, means that a sequence such as `B-D I-D O I-D O I-P B-S` yields two entity spans, one of the D type, spanning four subtokens, and one of the S type, spanning a single token. Such a heuristic forces proper disambiguation of B vs I tags, but is forgiving of O/I-tag errors. Finally, predictions of different types are separated for testing.

3 Synthetic Dutch data augmentation

In our experiments, we utilized the GPT4o model accessed via OpenAI API to generate synthetic data. To ensure deterministic and highly formatted outputs, the generation temperature was set to zero.

The prompt engineering process was designed carefully aiming at mitigating the inherent biases of the generative model. We specifically adapt the principles from the DisTEMIST guidelines for disease recognition (Miranda-Escalada et al., 2022), the MedProcNER guidelines for clinical procedures (López et al., 2023), and the SympTEMIST guidelines for symptoms and clinical findings (Lima-López et al., 2023). These

[2]Input sequences are prechunked to a maximum of 150 space-delimited words/tokens to prevent exceeding encoder maximum sequence length after subtokenization; we found the limit of 150 appropriate for all tested systems.

frameworks provide the rigorous conceptual boundaries necessary to keep generative models from hallucinating or misclassifying overlapping medical concepts as much as possible.

For the disease annotation task, the initial implementation revealed a strict disease bias where the model ignored structural pathologies and lifestyle risk factors. We refined the instructions to explicitly include structural defects like bone erosions and habits like smoking status, ensuring alignment with the DisTEMIST conventions. For the symptom annotation task, the instructions were designed to enforce contextual isolation, ordering the model to extract radiological findings and physical signs independently of any surrounding systemic diagnoses. Additionally, the procedure annotation prompt was modified to include a verb exception. Because clinical Dutch heavily utilizes past participles to describe procedural actions, the model was instructed to extract verbs when they functioned as the primary representation of a medical intervention.

The pipeline also incorporated sparse example prompting, where input texts and their gold standard outputs were provided within the system prompt, in order to ground the generative model and drastically reduce formatting errors. To respect request quotas, an automated pacing and retry mechanism was embedded in the processing script, pausing execution when rate limits were reached and preventing data loss.

Prompts are included in the Appendix.

4 Submitted Systems

We present five systems based on one of the three pretrained encoder models:

- medRoBERTa.nl[3], finetuned on the Dutch training data (`nl_med`) or the Dutch training data merged with synthetic annotations (`nl+synth_med`)
- robbert-2023-dutch-base[4], finetuned on the Dutch training data (`nl_robbert`)
- xlm-roberta-base[5], finetuned on Dutch training data (`nl_xlmrb`) or on all seven languages (`all_xlmrb`)

[3] https://huggingface.co/CLTL/medRoBERTa.nl

[4] https://huggingface.co/DTAI-KULeuven/robbert-2023-dutch-base

[5] https://huggingface.co/FacebookAI/xlm-roberta-base

All three pretrained encoder models have the same number of layers and layer sizes, but they differ by their vocabulary sizes and training. Whereas medRoBERTa.nl and RobBERT are trained with a vocabulary of 50k subtokens, xlm-roberta-base uses 250k subtokens, and is about twice as large in total parameter size.

5 Experiments

5.1 Setup

We used Adam for optimizing, tuning only the (constant) learning rate in the range 1e-5 to 1e-4. For tuning, we apply a 90/10% document-id-based split for training/validation. All submitted systems are finetuned with a learning rate of 1e-5. Training[6] is run over 20 epochs with a batch size of 32, keeping the last checkpoint.

5.2 Results

We did not optimize the model for the task in any specific way, such as adding a CRF layer or applying an ensemble approach. Using a standard architecture and training regime enables us to compare the strength of different models and annotations as factors. Hence, we did not aim to obtain the highest possible score given the test data but instead tried to learn about the general applicability of the classifiers to actual Dutch clinical notes. In the task our best system ranked 10th and our worst system ranked 22nd on precision metrics. The results for the detection of *diseases*, *procedures*, and *symptoms* are shown in Table 2.

Although the differences between our submissions are small, we can make the following observations:

- The XLM-RoBERTa model trained with the annotations for all languages (all_xlmrb) has the highest F1 for all three subtasks: *diseases*, *procedures* and *symptoms*.
- XLM-RoBERTa trained with projected Dutch annotations (nl_xlmrb) is the second best system, followed closely by medRoBERTa.nl trained with the same data (nl_med); medRoBERTa.nl performs better on the whole for *symptoms*.
- The general Dutch model RobBERT (nl_robbert) trained with projected Dutch

[6] Models are trained on the whole data, i.e. training and validation splits.

	strict				character			
	precision	recall	F1	rank	precision	recall	F1	rank
disease								
all_xlmrb	0.6221	0.6262	0.6241	14	0.7356	0.7391	0.7373	14
nl_xlmrb	0.6124	0.6148	0.6136	15	0.7277	0.7302	0.7290	16
nl_med	0.5829	0.5779	0.5804	16	0.7162	0.7146	0.7154	17
nl+synth_med	0.5347	0.5637	0.5488	19	0.6713	0.7058	0.6881	18
nl_robbert	0.5331	0.5488	0.5409	20	0.6655	0.6896	0.6773	19
procedure								
all_xlmrb	0.6556	0.5933	0.6229	12	0.7917	0.7153	0.7516	12
nl_xlmrb	0.6270	0.5898	0.6078	13	0.7672	0.7238	0.7449	13
nl_med	0.6044	0.5755	0.5896	15	0.7496	0.7199	0.7345	14
nl_robbert	0.5665	0.5874	0.5768	17	0.6956	0.7279	0.7114	16
nl+synth_med	0.5342	0.5658	0.5495	19	0.6744	0.7157	0.6944	18
symptom								
all_xlmrb	0.5510	0.4795	0.5128	12	0.7104	0.6156	0.6596	12
nl_med	0.5516	0.4303	0.4834	15	0.7267	0.5701	0.6389	14
nl_xlmrb	0.5482	0.4409	0.4887	16	0.7124	0.5723	0.6347	16
nl_robbert	0.5018	0.4171	0.4555	18	0.6721	0.5621	0.6122	19
nl+synth_med	0.4392	0.4025	0.4200	19	0.6160	0.5605	0.5870	20

Table 2: System scores by category, with F1 competition rank, ordered by character F1

annotations and medRoBERTa.nl trained with both projected and synthetic Dutch annotations (nl+synth_med) perform worse.

- models tend to reach better precision than recall for *procedures*, and even more so for *symptoms*

Although the model that uses the (projected) data from all languages (all_xlmrb) performed best, the improvement over monolingual Dutch training appears modest given the volume of training data.

For the Dutch-finetuned systems, XLM-RoBERTa performs best in the categories *disease* and *procedure*, and medRoBERTa.nl second; both systems perform comparatively for *symptoms*, the former exhibiting better recall but poorer precision. The smaller size of medRoBERTa.nl makes it the better model there, due to its lower computational cost. However, in general, XLM-RoBERTa better represents the data, gaining from a larger vocabulary and training data sizes. It remains to be seen how adaptable medRoBERTa.nl is to other medical texts than its training data. Further analysis could also consider the relation to the different types of entity.

Enriching training data with synthetic annotations generally led to the worst results, but most of the loss is incurred on precision. We expect that this is a result of the model being finetuned to recognize more entities, while being tested on data that are similar to the provided training data. When we engineered the prompts for the generative model, we explicitly instructed it to identify subtle edge cases. For instance, we told it to extract localized structural pathologies as diseases and routine diagnostic actions as procedures. It is highly probable that the model learned to extract a more exhaustive set of entities from the synthetic data, only to be penalized during testing against a conservatively annotated gold standard. Closer analysis would be required to evaluate the augmented data.

6 Conclusion

We have presented multiclass token classifying systems with the aim of comparing different training data and pretrained models for Dutch. We find that multilingual pretrained models perform best overall, also compared to a specialized, medical Dutch model. For future work, we would like to evaluate medRobBERTa.nl against the training notes in more detail, to shed light on the differences in data within a domain. We would also like to investigate further the differences between our synthetic annotations and the training annotations.

Limitations

Although pretrained encoder models are straightforward to apply to multiclass token classification, the multiclass setting forced us to discard some training annotations when they overlapped; we also do not know how much training data was lost by discarding unaligned sentences. Furthermore, reconstructing entity spans from BIO tags remains heuristic and may suffer in the context of long spans, as was the case here.

Acknowledgments

This research was sponsored by the Vrije Universiteit Amsterdam and through the NWO Spinoza-Vossen project "Understanding Language by Machines". We thank SURF (www.surf.nl) for the support in using the Dutch National Supercomputer Snellius.

References

Shutao Chen and Piek T.J.M. Vossen. 2026. A cheap lunch: Synthetic annotation with reduced human effort for medical text mining. In *Proceedings of the Fifteenth Language Resources and Evaluation Conference*, pages 10353–10364, Palma de Mallorca, Spain. ELRA Language Resource Association.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. Robbert: a dutch roberta-based language model. In *Findings of the association for computational linguistics: EMNLP 2020*, pages 3255–3265.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Fernando Gallego-Donoso, Salvador Lima-López, Judith Rosell, Eulàlia Farré-Maduell, and Martin Krallinger. 2026. The MultiClinAI Shared Task on Multilingual Clinical Corpus Construction and Concept Extraction: Systems, Evaluation, and Datasets. In *Proceedings of the 11th Social Media Mining for Health (#SMM4H) and Health Real-World Data (HeaRD) Workshop and Shared Tasks*. Association for Computational Linguistics.

Enshuo Hsu and Kirk Roberts. 2025. Leveraging large language models for knowledge-free weak supervision in clinical natural language processing. *Scientific Reports*, 15(1):8241.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified MRC framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859, Online. Association for Computational Linguistics.

Salvador Lima-López, Eulàlia Farré-Maduell, Laura Vigil-Giménez, Luis Gascó-Sánchez, and Martin Krallinger. 2023. Symptemist guidelines: Annotation and normalization for clinical symptoms, signs and findings.

Guillermo Lopez-Garcia, Jose Miguel Acitores Cortina, Jacob Berkowitz, Joey Chan, Ganesh Chandrasekar, Sumon Kanti Dey, Ivan Flores Amaro, Fernando Gallego, Lauren Gryboski, Ari Z Klein, Martin Krallinger, Salvador Lima-López, Tomohiro Nishiyama, Lisa Raithel, Ahmad Rezaie Mianroodi, Roland Roller, Judith Rosell, Frank Rudzicz, Abeed Sarker, and 8 others. 2026. Overview of the 11th Social Media Mining for Health (#SMM4H) and Health Real-World Data (HeaRD) Shared Tasks at ACL 2026. In *Proceedings of the 11th Social Media Mining for Health (#SMM4H) and Health Real-World Data (HeaRD) Workshop and Shared Tasks*. Association for Computational Linguistics.

Salvador Lima López, Eulàlia Farré Maduell, Luis Gascó Sánchez, and Martin Krallinger. 2023. Medprocner/proctemist guidelines: Annotation and normalization of clinical procedures in medical documents.

Antonio Miranda-Escalada, Luis Gascó, Salvador Lima-López, Eulàlia Farré-Maduell, Darryl Estrada, Anastasios Nentidis, Anastasia Krithara, Georgios Katsimpras, Georgios Paliouras, and Martin Krallinger. 2022. Overview of distemist at bioasq: Automatic detection and normalization of diseases from clinical texts: results, methods, evaluation and multilingual resources.

Vitor Oliveira, Gabriel Nogueira, Thiago Faleiros, and Ricardo Marcacini. 2025. Combining prompt-based language models and weak supervision for labeling named entity recognition on legal documents. *Artificial Intelligence and Law*, 33(2):361–381.

Stella Verkijk and Piek Vossen. 2025. Creating, anonymizing and evaluating the first medical language model pre-trained on Dutch Electronic Health Records: MedRoBERTa.nl. *Artificial Intelligence in Medicine*, 167:103148.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. 2025. GPT-NER: Named entity recognition via large language models. In *Findings of the Association for Computational Linguistics: NAACL*

2025, pages 4257–4275, Albuquerque, New Mexico. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Appendix

Prompt Used to Generate Synthetic Labels (Disease)

You are an expert medical annotator working with Dutch clinical texts. Your task is to extract mentions of DISEASES ONLY, strictly adhering to the adapted DisTEMIST annotation guidelines. You must NOT extract transient Symptoms, Signs, or Clinical Procedures.

DisTEMIST Annotation rules: 1. Strict entity filtering (diseases only): A 'Disease' is defined as a disorder in health with temporal persistence, but in this specific task, it also includes localized structural damages and specific risk factors. - Include Systemic Diseases: e.g. 'diabetes', 'matige acute pancreatitis', 'carcinoom'. - Include Structural or Localized Pathologies: Annotate persistent physical defects, tissue damages, and pathological formations (e.g. 'peri-implantaire pocket', 'botdefect'). - Include Habits or Risk Factors: By convention, annotate smoking status and similar pathological lifestyle markers (e.g. 'roker', 'niet-roker'). - Exclude Transient Symptoms or Signs: Do not annotate temporary clinical observations (e.g. do NOT extract 'pijn' [pain], 'bloeding' [bleeding]). - Exclude Procedures: Do not annotate treatments or surgeries. 2. Full specific spans: Extract the longest, most specific noun phrase for the disease. This includes relevant adjectives, severity modifiers, and anatomical locations (e.g. 'marginaal peri-implantair botdefect' rather than just 'botdefect'). 3. No negations in span: Do not include negation markers in the extracted string (e.g. for 'geen diabetes', extract ONLY 'diabetes'). *Exception: For established lifestyle habits like 'niet-roker' (non-smoker), the entire hyphenated term is the entity and must be extracted fully.* 4. No verbs: Do not extract verbs or verbal actions. Only extract noun phrases. 5. Abbreviations: Extract acronyms or abbreviations if they represent a disease (e.g. 'COPD', 'DM type 2'). 6. Exact match: Ensure the exact string matches the text perfectly, maintaining the original syntax and compound words.

Return the results ONLY as a valid JSON list of strings, representing the exact phrases as they appear in the text. Do not include any other text, markdown formatting, or explanations.

Prompt Used to Generate Synthetic Labels (Procedure)

You are an expert medical annotator working with Dutch clinical texts. Your task is to extract mentions of CLINICAL PROCEDURES ONLY, strictly adhering to the adapted MedProcNER (ProcTEMIST) annotation guidelines. You must NOT extract Diseases, Disorders, Symptoms, or Signs.

MedProcNER Annotation rules: 1. Strict Entity Filtering (Procedures ONLY): A 'Procedure' is an activity performed on a patient for diagnosis, treatment, therapy, prevention, or screening. - Include major interventions: Surgeries, imaging tests, and therapeutic interventions (e.g. 'echografie', 'bloedtransfusie'). - Include routine or diagnostic actions: Do not ignore standard medical activities (e.g. 'lichamelijk onderzoek' [physical exam], 'anamnese' [history taking], 'controle' [check-up], 'laboratoriumonderzoek'). - Include Therapeutic administration: The act of treating with a drug or therapy (e.g. 'behandeling met hydantoïne', 'vaccinatie'). - Exclude: Diseases and symptoms (e.g. do NOT extract 'diabetes', 'pijn', 'tumor'). 2. Contextual Isolation: Do not ignore routine procedures, tests, or standard treatments even if the text is overwhelmingly focused on describing a severe disease or its symptoms. Extract the procedures independently. 3. The Verb Exception (CRITICAL): While noun phrases are preferred, you MUST extract verbs or past participles if they are the primary representation of the medical procedure in the sentence (e.g. extract 'gehecht', geopereerd'). 4. Full Specific Spans: Extract the exact, complete phrase representing the procedure, including essential modifiers, anatomical targets, or equipment used (e.g. 'transfusie van 6 rode bloedcelconcentraten', 'correctie van de hernia inguinalis met marlex mesh'). 5. Specific Conventions (MedProcNER Rules): - Commercial Brands: Include commercial brand names if used synonymously with a procedural device or intervention (e.g. 'Ventimask', 'Babylog 8000 plus'). - State Changes or Therapeutic Concepts: Include complex therapeutic transitions (e.g. 'weaning', 'ontwenning',

'extubatie'). - Preparatory Procedures: Pre-surgical optimizations are procedures (e.g. 'optimalisatie van de anemie'). 6. NO Negations or Uncertainty in Span: Do not include negation or hypothetical markers. If a procedure was planned or not performed, extract ONLY the core procedure name. 7. Exact Match: Ensure the exact string matches the text perfectly, maintaining the original syntax and compound words in Dutch.

Return the results ONLY as a valid JSON list of strings, representing the exact phrases as they appear in the text. Do not include any other text, markdown formatting, or explanations.

Prompt Used to Generate Synthetic Labels (Symptom)

You are an expert medical annotator working with Dutch clinical texts. Your task is to extract mentions of SIGNS, SYMPTOMS, and CLINICAL FINDINGS ONLY, strictly adhering to the adapted SympTEMIST annotation guidelines. You must NOT extract Diseases or Disorders or Clinical Procedures.

SympTEMIST Anntation Rules: 1. Strict Entity Filtering: Filter out a transient observation, physiological state, physical sign, or radiological/morphological finding. - Include Symptoms/Signs: 'pijn bij palpatie' (pain), 'misselijkheid', 'koorts', 'syncope'. - Include Morphological & Radiological Findings: 'massa' (mass), 'tumor' (when described as a physical/visual finding), 'laesie' (lesion), 'verhoogde botdichtheid' (increased density), 'erosie' (erosion). - Exclude Diseases: Diagnosed persistent conditions (e.g. 'diabetes', 'sclerose', 'ameloblastoom'). - Exclude By Convention: 'adenopathie', 'anemie', 'atelectase', 'effusie/derrame', 'ulcus', and 'cyste/pseudocyste' are considered DISEASES. Do NOT extract them here. 2. Contextual Isolation (CRITICAL): Do not ignore valid symptoms, signs, or radiological findings just because they are caused by or surrounded by a severe underlying disease. Extract the findings independently. 3. Span Guidelines: Extract the exact, concise noun phrase representing the symptom/finding along with essential anatomical modifiers (e.g. 'tumor in de basale mandibulaire regio', 'erosie van de cortex'). - AVOID overly long, non-inherent descriptions. 4. No Negations in Span: Do not include negation markers. If a symptom is negated, extract ONLY the symptom name. 5. No Verbs: Do not extract verbs. Only extract noun phrases. 6. Exact Match: Ensure the exact string matches the text perfectly, maintaining the original syntax in Dutch.

Return the results ONLY as a valid JSON list of strings, representing the exact phrases as they appear in the text. Do not include any other text, markdown formatting, or explanations.