

RESEARCH

Open Access



# Negation detection in Dutch clinical texts: an evaluation of rule-based and machine learning methods

Bram van Es<sup>1,8\*</sup>, Leon C. Reteig<sup>2</sup>, Sander C. Tan<sup>3</sup>, Marijn Schraagen<sup>4</sup>, Myrthe M. Hemker<sup>5</sup>, Sebastiaan R. S. Arends<sup>6</sup>, Miguel A. R. Rios<sup>7</sup> and Saskia Haitjema<sup>1</sup>

\*Correspondence:  
b.vanes-3@umcutrecht.nl

<sup>1</sup> Central Diagnostic Laboratory, University Medical Center Utrecht, Utrecht University, Utrecht, The Netherlands

<sup>2</sup> Center for Translational Immunology, University Medical Center Utrecht, Utrecht, The Netherlands

<sup>3</sup> Department for Research & Data Technology, University Medical Center Utrecht, Utrecht, The Netherlands

<sup>4</sup> Institute for Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands

<sup>5</sup> Utrecht Institute of Linguistics OTS & Department of Languages, Literature and Communication, Utrecht University, Utrecht, The Netherlands

<sup>6</sup> Department of Medical Informatics, University of Amsterdam, Amsterdam, The Netherlands

<sup>7</sup> Centre for Translation Studies, University of Vienna, Vienna, Austria

<sup>8</sup> MedxAl, Amsterdam, The Netherlands

## Abstract

When developing models for clinical information retrieval and decision support systems, the discrete outcomes required for training are often missing. These labels need to be extracted from free text in electronic health records. For this extraction process one of the most important contextual properties in clinical text is negation, which indicates the absence of findings. We aimed to improve large scale extraction of labels by comparing three methods for negation detection in Dutch clinical notes. We used the Erasmus Medical Center Dutch Clinical Corpus to compare a rule-based method based on ContextD, a biLSTM model using MedCAT and (finetuned) RoBERTa-based models. We found that both the biLSTM and RoBERTa models consistently outperform the rule-based model in terms of F1 score, precision and recall. In addition, we systematically categorized the classification errors for each model, which can be used to further improve model performance in particular applications. Combining the three models naively was not beneficial in terms of performance. We conclude that the biLSTM and RoBERTa-based models in particular are highly accurate in detecting clinical negations, but that ultimately all three approaches can be viable depending on the use case at hand.

**Keywords:** Natural language processing, Text mining, Negation detection

## Introduction

The increasing availability of clinical care data, affordable computing power, and suitable legislation provide the opportunity for (semi-)automated decision support systems in clinical practice. An important step in the development of such a decision support system is the accurate extraction of relevant labels to train the underlying models. These labels are rarely directly available as structured data in electronic health records (EHRs)—and even if they are, they often lack the precision and reliability [41] for a clinical decision support system. Therefore, extraction of labels from free text in the EHR—which contains the richest information and the appropriate amount of nuance—is needed.



© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

To this end, we need to consider the context in which medical terms are mentioned. One of the most important contextual properties in clinical text is *negation*, which indicates the *absence* of findings such as pathologies, diagnoses, and symptoms. As they make up an important part of medical reasoning, negations occur frequently: one study estimated that more than half of all medical terms in certain clinical text are negated [6]. Accurate negation detection is critical when labels and features from free text in the EHR are extracted for use in clinical prediction models. But improving information retrieval through negation detection has many other use cases in healthcare, including administrative coding of diagnoses and procedures, characterizing medication-related adverse effects, and selection of patients for inclusion in research cohorts.

Negation detection is not a trivial task, due to the large variety of ways negations are expressed in natural language<sup>1</sup>. It can either be performed with a rule-based approach or through machine learning. In this paper, we evaluate the performance of one rule-based method (based on ContextD [1]) and two machine learning methods (a bidirectional long-short term memory model implemented in MedCat [25], and a RoBERTa-based [28] Dutch language model) for detection of negations in Dutch clinical text.

In their simplest form, traditional rule-based methods consist of a list of regular expressions of negation triggers (e.g. “no evidence for”, “was ruled out”). When a negation trigger occurs just before or after a medical term in a sentence, the medical term is considered negated. Examples include NegEx [5], NegFinder [33], NegMiner [13] and ConText [19, 38]. Some approaches also incorporate the grammatical relationships between the negation and medical terms. For example, incorporating part-of-speech tagging to determine the noun phrases that a negation term could apply to (see e.g. NegExpander [3]), or using dependency parsing to uncover relations between words (see e.g. NegBio [35], negation-detection [16], DepNeg [40] and DEEPEN [31]). Moreover, distinguishing between the different types of negations (syntactic, morphological, sentential, double negation) as well as adding word distance has been proven helpful (see e.g. NegAIT [32] and Slater et al. [39]). While usually tailored for English, some of these methods have been adapted for use in other languages, including French [11], German [8] and Spanish [7, 9], as well as Dutch [1].

The main *advantages* of rule-based negation detection methods are that they are transparent and easily adaptable and do not require any scarce labeled medical training data. Rule-based methods can be surprisingly effective. Goryachev et al. [17] demonstrate that the relatively simple NegEx can be more accurate than machine learning-based methods, such as a Support Vector Machine (SVM) trained on the part-of-speech-tags surrounding the term of interest.

The main *disadvantage* of rule-based methods is that they are by definition unable to detect negations that are not explicitly captured in a rule. Depending on the use case, this can severely hamper their performance. This is where machine learning methods come into play, as they may outperform rule-based methods by picking up rules implicitly from annotated data.

---

<sup>1</sup> For example, the rule-based system used in this work contains nearly 400 different patterns of expressing negation.

One such machine learning method is the bidirectional long-short term memory model (biLSTM), a neural network architecture that is particularly suited for classification of sequences such as natural language sentences. This model processes all words in a sentence sequentially, but in contrast to traditional neural network methods, a biLSTM takes the output of previous words into account to model relations between words in the sentence. For a biLSTM model the processing is bidirectional, meaning that sentences are processed in the (natural) forward direction, as well as the reverse direction.

Based on a conventional biLSTM (see e.g. Graves and Schmidhuber [18]), Sun et al. [42] developed a hybrid biLSTM Graph Convolutional Network (biLSTM-GCN) method. The biLSTM can also be combined with a conditional random field [20]. Other machine learning approaches include an SVM that has access to positional information of the tokens in each sentence Cruz Díaz et al. [10], and the use of conditional random fields in the NegScope system by Agarwal and Yu [2].

A more recent machine learning model is RoBERTa [28], a bidirectional neural network architecture that is pre-trained on extremely large corpora using a self-supervised learning task, specifically to fill in masked tokens. This masking does not require external knowledge as the selection of tokens to be masked can be performed automatically. RoBERTa is part of a family of models, which primarily vary in learning task, that are based on the transformers architecture [43]. Once pre-trained, a transformer model can be finetuned with supervised learning tasks for e.g. negation detection or named entity recognition. Lin et al. [27] show that a zero-shot language model such as BERT performs well on the negation task and does not require domain adaptation methods. Khandelwal and Sawant [23] developed NegBERT, a BERT model finetuned on open negation corpora such as BioScope [46].

The goal of the current paper is to compare the performance of rule-based and machine learning methods for Dutch clinical data. We conduct an error analysis of the types of errors the individual models make, and also explore whether combining the methods through ensembling offers additional benefits. Python implementations of all the evaluated models are available on GitHub.<sup>2</sup>

## Data

We used the Erasmus Medical Center Dutch clinical corpus (DCC) collected by Afzal et al. [1] (published together with ContextD) that contains 7490 Dutch anonymized medical records annotated with medical terms and contextual properties. All text strings that exactly matched an entry in a subset of the Dutch Unified Medical Language System (UMLS, [4]) were considered a medical term. These medical terms were subsequently annotated for three contextual properties: temporality, experiencer, and negation. In this paper we focus on the binary context-property negation. The label *negated* was given when evidence in the text was found that indicated that a specific event or condition did not take place or exist, otherwise the label *not negated* was assigned.

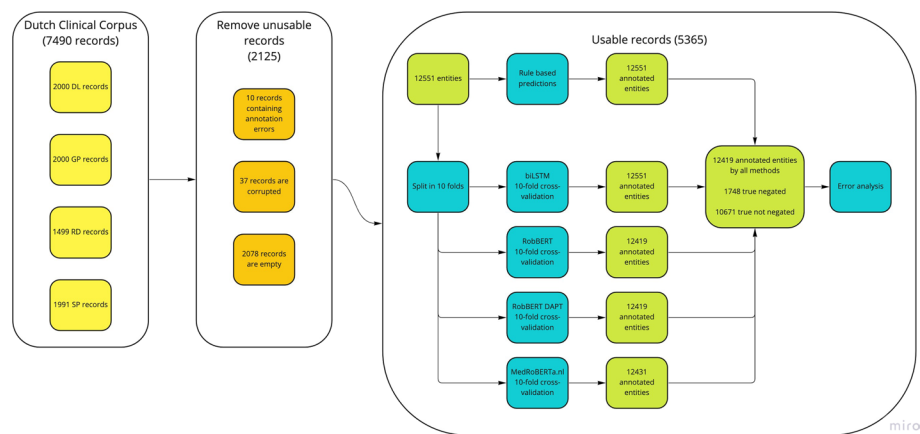
As illustrated in Fig. 1, we excluded 2125 records in total from further analysis, primarily because no annotation was present (this was the case for 2078 records),

---

<sup>2</sup> <https://github.com/umcu/negation-detection/releases/tag/v1.0.0>.

**Table 1** Basic textual statistics of the selected DCC records, showing the mean value per record and the boundaries of the second and third quartile (top) and the total count in the dataset (bottom)

Letter category	# sentences	# words	# unique words	Word length
General Practitioner entries	2.1 (1, 2) 2034	17.8 (8, 23) 33840	16.8 (8, 22) 11080	5.9 (5, 6.5)
Specialist letters	3.8 (2, 4) 2737	30.0 (9, 34) 29674	24.9 (8, 30) 10207	7 (5.8, 7.7)
Radiology reports	3.6 (2, 4) 3939	19.1 (7, 26) 28614	16.8 (6, 23) 6371	7.2 (6.1, 8)
Discharge letters	4.1 (2, 5) 3057	33.8 (14, 45) 33458	27.8 (13, 38) 6351	6.1 (5.5, 6.6)

**Fig. 1** Data flow diagram

otherwise the file containing the source text or its annotation was corrupted (37 records), or the annotation did not correspond to a single medical term (10 records, e.g. only a single letter was annotated, or a whole span of text containing multiple medical terms). This left 5365 usable records for analysis, containing a total of 12551 annotated medical terms. A small number of medical terms were not processed by the RoBERTa-based models, because of the imposed maximum record length of 512 tokens in our implementation. We excluded these medical terms from analysis with other methods as well, resulting in a final set of 12419 annotated medical terms.

The corpus consists of four types of clinical records, which differ in structure and intent (for details, see Afzal et al. [1]). Basic statistics are presented in Table 1 and a representative example of each record type, including various forms of negation, is provided in Supplementary Material A.

## Methodology

We employed three distinct methods to identify negations: a rule-based approach called ContextD, a biLSTM from MedCAT and a fine-tuned Dutch RoBERTa model. These methods are cross-validated using the same ten folds.

### Rule-based approach

The rule-based ConTextD algorithm [1] is a Dutch adaptation of the original ConText algorithm [19].

The backbone of the ConText algorithm is a list of (regular expressions of) negation terms (“negation triggers”). A given medical term is considered to be negated when it falls within the scope of a negation trigger. The default scope in ConText is the remainder of the sentence after/before the trigger. Each negation trigger has either a forward (e.g., “no evidence of ...”) or backward scope (e.g., “... was ruled out”).

ConText has two more types of triggers aside from negation triggers: pseudo-triggers and termination triggers. Pseudo-triggers are phrases that contain a negation trigger but should not be interpreted as such (e.g., “not only” is a pseudo-trigger for the negation trigger “not”). Pseudo-triggers take precedence over negation triggers: when a pseudo-trigger occurs in a sentence, the negation trigger it encompasses is not acted upon. Termination triggers serve to restrict the scope of a negation trigger. For example, words like “but” in the sentence “No signs of infection, but pneumonia persists”, signal that the negation does not apply to the entire sentence. Using “but” as termination trigger prevents the algorithm to consider “pneumonia” to be negated while “infection” is still considered negated.

We used the Dutch translation of the original triggers from ConText, as produced by the ContextD authors.<sup>3</sup> These triggers were used in conjunction with MedSpaCy<sup>4</sup> [14], a Python implementation of the ConText algorithm. Because ConText defines the scope of a negation trigger in number of words or the boundary of the sentence (the default), raw text also needs to be tokenized and split into separate sentences. We used the default tokenizer and the dependency-parser-based sentence splitter of the `nl_core_news_sm-2.3.0` model in spaCy<sup>5</sup>, a generic Python library for NLP.

### MedCAT’s biLSTM

The open-source multi-domain clinical natural language processing tool Medical Concept Annotation Toolkit (MedCAT) incorporates Named Entity Recognition (NER) and Entity Linking (EL), to extract information from clinical data [25]. MedCAT contains a component named MetaCAT for dealing with context properties after a concept is identified in a clinical text. MetaCAT makes use of bidirectional Long-Short Term Memory networks (biLSTMs) [18]. This sequence encoding can be combined directly with a classifier such as a fully connected network (FCN) or indirectly via a Conditional Random Field or Graph Convolutional Network to facilitate interaction between entities. In this work we use an FCN for the final classification. By MetaCAT default, the biLSTM observes 15 tokens to the left and 10 tokens to the right of the annotated concept.

MetaCAT replaces target terms, for example *diabetes mellitus*, with an abstract placeholder, such as *[disease]*. This allows the model to learn a shared representation between concepts, which increases the generalization behavior of the model for negation detection. For example, the model now only needs to learn that *no signs of [disease]* is a

---

<sup>3</sup> Erasmus Medical Center website.

<sup>4</sup> medspacy, version 0.1.0.2.

<sup>5</sup> spaCy, version 2.3.5.

negation, instead of learning *no signs of diabetes mellitus, no signs of bronchitis, ...* for each disease separately.

MetaCAT uses a Byte Pair Encoding (BPE) tokenizer which operates on a subword level [15]. Earlier research [29] showed that the subword tokenizer outperforms traditional tokenizers that operate on word level. When training MetaCAT's biLSTM, embeddings are required. For this reason, first a set of Continuous Bag of Words (CBOW) embeddings were created using a set of Dutch Wikipedia articles in the medical domain. Secondly, the biLSTM model is trained on the annotated DCC data using the default MetaCAT settings.

While MedCAT provides a full pipeline including NER and entity linking, in the current project we only use the biLSTM negation classifier component, using the entities marked in the DCC as input. This allows for easy comparison with the other two methods that do not share the same pipeline architecture.

### RoBERTa

RoBERTa [28] is part of a family of language models built on top of the transformers architecture that allows for learning general contextual representations using self-supervised training. The variation in models such as BERT [22], XLM [26] and T5 [37] primarily comes from differences in training objectives.

A benefit of RoBERTa, BERT and similar transformer-based methods is that the context of a term can play an important role. With sequential models such as LSTMs long range inter-word dependencies are hardly taken into account for the simple reason that direct neighbors have more influence on each other<sup>6</sup>. Bidirectional LSTM's alleviate the uni-directionality but still suffer from the inability to incorporate large contexts.

A possible downside of using a pre-trained model is that the standard version of RoBERTa is trained on general text, which may not perform well in the medical domain.

The RobBERT language model [12] is a RoBERTa architecture pretrained on the Dutch SoNaR corpus (see Oostdijk et al. [34]) using Masked Language Modeling, a self-supervised learning process where the model learns to fill in masked tokens. SoNaR consisted primarily of texts obtained from publicly available media. MedRoBERTa.nl [44] is a RoBERTa model that was trained from scratch on 11.8GB of EHRs from the Dutch AUMC tertiary care center.

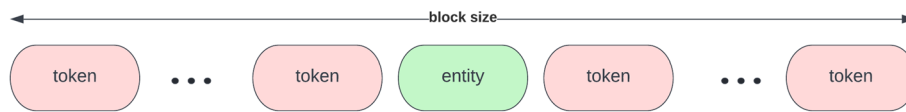
Following Lin et al. [27], we compared RobBERT with a domain-adapted (DAPT) RobBERT. Here the domain-adaptation is performed by continued pre-training of the model on a domain-specific corpus consisting of Dutch Medical Wikipedia, NHG directives and standards<sup>7</sup> (medical guidelines for general practitioners), Richtlijndatabase<sup>8</sup> (medical guidelines for hospital specialties) and Huisarts en Wetenschap<sup>9</sup> (monthly magazine of the Dutch general practitioners' association, issues 1957–2019) (see Additional file 1: Table S3 for more detail).

<sup>6</sup> Recurrent neural networks suffer from the exploding and vanishing gradient problem, with LSTMs and GRUs this is resolved basically through gates between sequence elements and weight clipping. These interventions allow for larger sequences but cannot prevent that there is a monotonic decline in influence away from each token.

<sup>7</sup> <https://www.nhg.org/>.

<sup>8</sup> <https://demedischespecialist.nl/>.

<sup>9</sup> <https://www.henw.org/>.



**Fig. 2** Variable size token window

We finetuned all RoBERTa models on the annotated DCC data for 3 epochs with a batchsize of 32 and 64 samples. We performed the finetuning in two ways; all layers are updated or only the final classification layer is trained for the negation detection task. The latter is also called self-supervised plus simple fit (or SSS).

Finally we considered the effect of decreasing the maximum number of tokens, i.e. the maximum sequence length. Decreasing the maximum sequence length will linearly decrease both the space and time complexity of the model training. We varied the number of tokens by taking a token window around each entity (i.e. an annotated medical term in the DCC) (Fig. 2). For the training this means that entities may occur in multiple token windows per record. For the validation we only use the entities in the center of each token window.

#### Ensemble classification

To investigate whether the three models could complement each other, we created an ensemble classifier. The classifier used a majority voting ensemble, which assigns the label that is predicted by at least two of the three classifiers.

#### Model evaluation

To gauge the performance of the models we applied 10-fold cross-validation. We inspected the errors in the test folds and assigned error categories facilitating inter-model comparison (see Sect. ). This setup also allows for easy evaluation of the proposed ensemble classifier post-hoc based on the predictions of the individual methods.

We evaluated each model's recall, precision, and F1-score for the *negated* class. Recall (true positive rate or sensitivity) is defined as the proportion of true positives out of the sum of true positives and false negatives, and thus indicates what proportion of all negations a model correctly identified. Precision (positive predictive value) is defined as the proportion of true positives out of all positives, and thus indicates how often the model was correct when it predicted that a term was negated. The F1-score is the harmonic mean of precision and recall.

#### Error analysis

To better understand the types of misclassifications that the models made, we reviewed all the false positives and false negatives separately for each of the three models, similar to what [1] did to evaluate ContextD. A false negative occurred when the model predicted “not negated” when a negation was present; a false positive occurred when the model predicted “negated” when no negation was present.

Error analysis allows to determine the expected gain of further steps aimed at preventing errors, which can be either (1) to apply preprocessing to remove artifacts that negatively influence model performance, (2) to remove inconsistencies in data



**Table 2** Definitions and examples of the error categories used for error analysis. Negation terms are underlined; the annotated medical terms are in brackets. Examples translated from the Dutch source text

Category	Definition and example
Uncommon negation	Negation term rarely occurs in the data ... which <u>argues against</u> a [diagnosis] of RA
Mminus	In clinical notes, a minus directly following a term indicates negation <i>pale-</i> , [nauseous] -, clammy+
Scope	Several medical terms follow/precede a negation, but the negation does not apply to all of them. This often occurs in a list, or simply when the negation occurs much later/earlier in the sentence <u>no</u> abdominal pain, [abrasion on leg]
Punctuation	Punctuation that is likely to hinder sentence splitting and/or correct recognition of the (scope of the) negation term <u>no</u> evid. for [aneurysm]
Negation of different term	The negation applies to another term close to the medical term <u>no</u> further investigation of [weight loss]
Wrong modality	The context is not negation, but hypothetical, historical, or otherwise <i>indication: to rule out</i> [osteopenia]
Speculation	The clinician expressed uncertainty instead of outright negation <u>no</u> [eczema] after all?
Ambiguity	The grammatical structure makes it unclear whether a term is negated or not ... <u>without</u> loss of function and [concussion] following...
Other	The type of error does not fit into any of the other categories
Annotation error	The original annotator assigned the wrong label

*Italic: paraphrase of example clinical text*

annotation and/or use synthetic data to train the models, or (3) to combine the different models in a complementary fashion.

We first categorized all errors made by a single model. We then attempted to reuse these categories as much as possible for the error analysis of the remaining two models, only creating new categories when there was a clear need. Each model's errors were reviewed by a single individual (BvE, LCR, and MS). This initial round was followed by two more rounds of discussion among the three reviewers, lumping and splitting the candidate categories, to obtain a final set of categories that was shared across models. This process resulted in 10 different error categories (Table 2). To assess the degree of consistency in categorization of the three reviewers, we computed Cohen's kappa coefficient on the subset of errors that were shared by multiple models.

## Results

We consider the model performance quantitatively by looking at overall performance metrics, and more qualitatively by analyzing and categorizing the errors that each model made.

### Overall performance

From the 12419 medical terms in 5365 medical records, 1748 medical terms were marked as negated by the annotators. Of these, 1687 concepts were identified by at least one of the negation detection models.

The precision, recall and F1 score for each negation detection method are reported in Table 3. RobBERT achieved the highest scores overall, followed by the ensemble method for a few metrics and record types.



**Table 3** Classification results across methods and data sources

Letter category	Prediction method	Precision	Recall	F1
Discharge letters	Rule-based	0.893	0.921	0.906
Discharge letters	BiLSTM	0.957	0.931	0.944
Discharge letters	RobBERT	0.953	<b>0.974</b>	0.963
Discharge letters	Voting ensemble	<b>0.963</b>	0.966	<b>0.964</b>
General Practitioner entries	Rule-based	0.674	0.801	0.732
General Practitioner entries	BiLSTM	0.889	0.889	0.889
General Practitioner entries	RobBERT	<b>0.950</b>	<b>0.912</b>	<b>0.931</b>
General Practitioner entries	Voting ensemble	0.930	0.886	0.908
Radiology reports	Rule-based	0.901	0.966	0.932
Radiology reports	BiLSTM	0.933	0.934	0.933
Radiology reports	RobBERT	<b>0.960</b>	0.963	<b>0.961</b>
Radiology reports	Voting ensemble	0.955	<b>0.965</b>	0.960
Specialist letters	Rule-based	0.807	0.840	0.823
Specialist letters	BiLSTM	0.922	0.835	0.876
Specialist letters	RobBERT	0.934	<b>0.890</b>	<b>0.911</b>
Specialist letters	Voting ensemble	<b>0.937</b>	0.862	0.898
All letters	Rule-based	0.825	0.892	0.857
All letters	BiLSTM	0.926	0.901	0.913
All letters	RobBERT	<b>0.951</b>	<b>0.938</b>	<b>0.944</b>
All letters	Voting ensemble	0.948	0.924	0.936

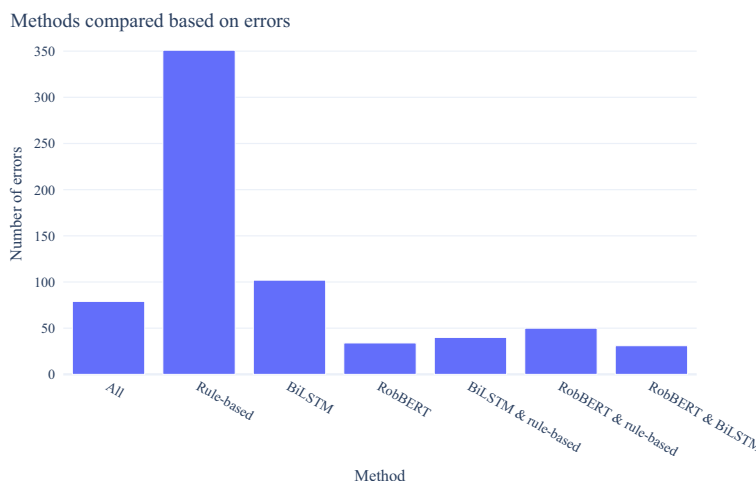
Bold: best score for a category of clinical notes

### **Rule-based**

While the performance of the rule-based method here was indeed comparable to the original (see the results in Table 5 of Afzal et al. [1]), some differences can be identified, that probably arise because we do not use exactly the same rules nor exactly the same dataset. First, there are two different variations of ContextD: the “baseline” rules, which were simply a translation of the original Context method [19], and the “final” rules, which were iteratively adapted using half of the dataset. Our set of rules is most similar to the “baseline” method, as we chose not to implement any of the modifications in the “final method” described in the paper. However, our set of rules likely still contains some elements of the “final method”. Second, while the original ContextD method was evaluated on only half the dataset (as the other half was used for finetuning the rules), we use the full dataset. In our evaluation, performance varied quite strongly over the folds, which indicates that the exact evaluation set that was used influenced the obtained results. Performance varied particularly strongly for the GP entries, which would also explain why the difference in performance with ContextD is largest for this category.

### **Machine learning (biLSTM, RobBERT)**

The rule-based method is outperformed by both machine learning methods in almost all cases. Its performance varies strongly over the different record types: it performs worst for the least structured records, particularly the GP entries, and best for the most structured records, particularly the radiology reports. The performance gap between the rule-based and machine learning methods shows the same pattern: the less structured the record, the larger the gap.



**Fig. 3** Number of misclassifications for different model combinations. The y-axis shows the number of errors in all possible intersections of the error sets made by the different models. That is, “All” is the number of entities that are misclassified by *all three* models; “Rule-based” is the number of entities that are misclassified by *only* the rule-based model; “RobBERT & BiLSTM” is the number of entities misclassified by *both* the RobBERT and BiLSTM models, *but not* the rule-based model; etc

	Oth	Ann	Spec	Amb	Sc	Pct	Dif	Mod	Minus
Uncommon	11	6	7	5	6	6	1	6	2
Other		2	7	3	3	7	4	0	0
Annotation			3	5	3	1	2	0	0
Speculation				13	3	1	8	3	0
Ambiguous					2	1	4	1	0
Scope						3	3	0	1
Punctuation							1	0	0
Different term								0	1
Modality									0

**Fig. 4** Confusion matrix of inter-annotator disagreement

The BiLSTM model consistently outperformed the rule-based approach, except for the radiology reports category, where performance was approximately equal. In turn, RobBERT outperformed the BiLSTM model with a difference of 0.02–0.05 in F1 score across record categories.

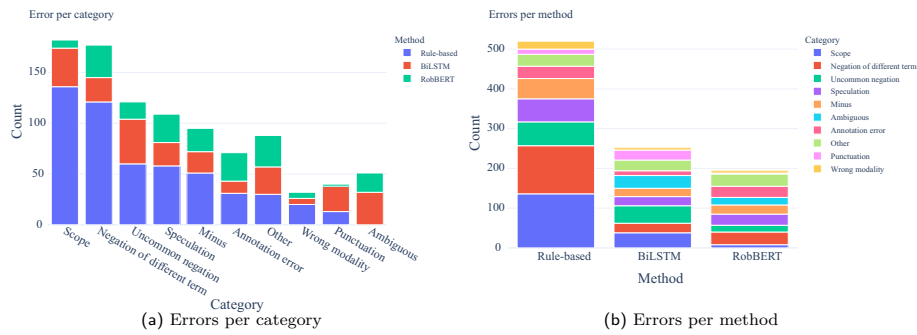
Additionally, we saw no consistent differences between different RobBERT implementations. The smallest 32-token window resulted in a slightly reduced accuracy, but drastically reduced computational resources (see Additional file 1 Tables S1 and S2).

Note that for the RobBERT and the BiLSTM models we did not apply threshold tuning to optimize for precision or recall—we simply took the default threshold of 0.5 (also note that calibration is required if such a probabilistic measure is applied in clinical practice).

**Model ensemble**

The RobBERT method outperforms the other models as well as the ensemble method when scored on the complete dataset. On the individual categories the ensemble method performs worse or similar to the RobBERT method.

Figure 3 shows that the RobBERT method (all bars with “RobBERT”) makes fewer errors than the voting ensemble (all bars with more than two methods). In particular,



**Fig. 5** Error category frequencies

the number of errors committed by RobBERT alone is smaller than the number of errors that are introduced when adding the BiLSTM and the rule-based method to the ensemble.

### Error analysis

We obtained a Cohen’s Kappa score of 0.48 when annotating agreement on the error category. This is considered as moderate agreement [30]. The categories involved in disagreement are shown in Fig. 4. The *uncommon negation* meta-category is a source of disagreement, showing that some annotators consider a negation uncommon while others choose a semantic error category. Also *other* as a (semantic) catch-all category is responsible for disagreement. From the specific categories the *speculation* label is most often subject of disagreement.

The moderate agreement is not surprising given that the error cases represent challenging annotations. It is important to note that it is possible that multiple categories apply for the same error due to model-specific interpretations, which would have a negative effect on our perceived inter-annotator agreement.

We should note that the *speculation* category is somewhat domain specific: a clinician might have observed that a particular diagnostic test yielded no indications for a particular diagnosis; arguably this is not a negation of the diagnosis but merely the explicit absence of a confirmation. These signals may or may not be considered as negations, depending on, for example, whether a test is seen as conclusive or if instead additional testing is required for establishing a diagnosis. This reasoning would require the integration of external knowledge, for example through UMLS. More broadly, the dichotomisation of negation/not-negated is perhaps too coarse given the high prevalence of explicitly speculative qualifications in electronic health records. One clear issue with the dichotomy not-negated/negated is that it biases the annotations and models towards the *non-negated* class, because the *negated* label requires explicit negations whereas *non-negated* is everything else, i.e., negations are more strictly constrained. In some cases it is beneficial if the bias is reversed, for instance to obtain affirmations with low false positivity. A mitigation of the non-negation bias is to introduce a model specifically for affirmations/non-affirmations, or indeed a separate label for speculation (see, e.g. Vincze [45]).

**Table 4** Overview of error categories per model

	Rule-based		biLSTM		RobBERT	
<i>False positives</i>						
Ambiguous	0	0%	14	16%	11	13%
Annotation error	11	3%	5	6%	13	15%
Minus	0	0%	8	9%	0	0%
Negation of different term	119	36%	24	27%	32	38%
Other	1	0%	13	14%	16	19%
Punctuation	0	0%	5	6%	1	1%
Scope	136	41%	6	7%	0	0%
Speculation	50	15%	10	11%	8	9%
Uncommon negation	0	0%	5	6%	0	0%
Wrong modality	14	4%	0	0%	4	5%
Total	331		90		85	
<i>False negatives</i>						
Ambiguous	0	0%	18	11%	8	7%
Annotation error	20	11%	7	4%	15	14%
Minus	51	27%	13	8%	23	21%
Negation of different term	2	1%	0	0%	0	0%
Other	29	15%	14	9%	15	14%
Punctuation	13	7%	20	12%	1	1%
Scope	0	0%	32	20%	8	7%
Speculation	8	4%	13	8%	20	18%
Uncommon negation	60	32%	39	24%	17	16%
Wrong modality	6	3%	6	4%	2	2%
Total	189		162		109	

**Model-independent issues**

The distribution of error categories over methods is shown in Fig. 5 and Table 4. The categories *annotation error*, *speculation*, *ambiguous* (together around 20–25% across models) may benefit from more specific annotation guidelines. The other errors can be classified as actual mistakes by the model. From these, preprocessing can potentially improve scope and punctuation-related errors. Modality can be improved using special-purpose classifiers similar to the current negation detection classifier. The remaining problematic errors are negation of a different term, uncommon negation, minus, and other errors (around 50% of all errors across models).

**Annotation- and uncertainty-related errors**

A significant amount of annotation errors was found, both for false positives and false negatives. This is consistent with Afzal et al. [1], who report that about 8% of the false positive negations were due to erroneous annotations (no percentage was reported for false negatives). These errors can be solved by improving the annotation, either through better guidelines or by more strict application and post-hoc checking of these guidelines.

Note that annotation errors will also be present in true negatives/positives, which will remain undetected in case the models make the same mistake as the annotators. If these annotation errors are randomly distributed this is a form of label smoothing, and as such the errors could be useful to reduce overfitting. However, it is likely that these annotation errors are not random and are indicative of inherent ambiguity.

The *speculation* and *ambiguous* categories (together around 20–25% across models) both stem from uncertainty-related issues, either expressed by the clinician (*speculation*) or in the interpretation of the text (*ambiguous*). These may present problems both during annotation and during model training, as the examples do not fully specify a negation, yet the intended meaning can often be inferred. More specific annotation guidelines could reduce these issues to some extent, by ensuring that the examples of a certain category are consistently annotated. However, the models may still be unable to capture such inferences, even if the examples are consistently annotated.

Typical examples for speculation are: *There is no clear [symptom]* or *The patient is dubious for [symptom]*. The English corpora BioScope [46], GENIA [24] and BioInfer [36] include uncertainty as a separate label, which may be beneficial for the current dataset as well.

#### Remaining errors

The other categories were more syntactic in nature. The word-level syntactic errors *scope*, *negation of different term*, and *uncommon negation* occur across methods. Other errors are due to the use of a *minus* to indicate negation or usage of colon and semicolon symbols (*punctuation*). The minus sign is however also used as a hyphen (to connect two words), which complicates handling of this symbol both in preprocessing and during model training.

As an example of possible mitigation measures, the following sentence produced a false negative (i.e., a negated term classified as non-negated) for the target term *redness*:

Previously an antibiotics treatment was administered(no redness).

In this example the negation word *no* is concatenated with an opening parenthesis and the previous word, which poses problems for tokenization. Such errors might be avoided by inserting whitespace during preprocessing.

The following sentence shows a false positive from the biLSTM classifier for the term *earache*:

since 1 day pain in the right lower lobe andcoughing, mucus, temp to38.2, pulmones no abnormality earache and deaf, oam right?

This is a scoping error, where the negation on the term “abnormality” is incorrectly extended to the target term “earache”. This issue would be difficult to correct in preprocessing, as it necessitates some syntactic and semantic analysis to normalize the sentence. This would closely resemble the processing by the rule-based system, therefore this is an example where model ensembling could be beneficial. However, given the multiple other textual issues in this sentence (such as missing whitespace, punctuation, and capitalization) a more robust alternative solution might be to maintain a certain standard of well-formedness in reporting, either through automatic suggestions, reporting guidelines, or both.

### **Rule-based**

Almost half of the false positives for the rule-based method fell under the “scope” category (41%, see Table 4). The default scope for a negation trigger extends all the way to the start (backward direction) or end (forward direction) of a sentence. For long sentences, or when sentences are not correctly segmented, the negation trigger may then falsely modify many medical terms. This occurred particularly often for short and unspecific negation triggers, such as *no* (Dutch: *niet, geen*). Potential solutions include improving sentence segmentation, restricting the number of medical terms that a single negation trigger can modify, restricting the scope to a fixed number of tokens (the solution used by the ContextD “final” algorithm for certain record types), or restricting the scope by adding termination triggers (the ContextD “final” algorithm added punctuation such as colons and semicolons as termination triggers for some record types). We determined that 32 out of the 140 false positives were caused by a missing termination trigger; adding just the single trigger *wel* (roughly meaning *but*) would have prevented 18 errors (5%).

Most other false positives were due to “negation of a different term”. These are perhaps more difficult to fix, but in some cases these could be prevented by adding pseudo-triggers that for instance prevent the trigger “no” from modifying a medical concept when followed by another term (e.g., “no relationship with”).

For the false negatives, the majority are caused by “uncommon negations”, i.e. negation triggers that were missing from the list of rules. A special case that caused a lot of errors was the minus (hyphen) symbol, which in clinical shorthand is often appended to a term to indicate negation. Other missing negation triggers that occurred relatively often were variations on *negative* (n=18, such as *neg*), *not preceded by* (n=8, e.g., *niet voorafgegaan door*), and *argues against* (n=5, e.g., *pleit tegen*). The obvious way to remedy these error categories is to simply add these negation triggers to the rule list, but this may introduce new problems. For instance, adding “-” as a negation trigger (as was done in the ContextD “final” algorithm) would negate any word that occurs before a hyphen (e.g., “infection-induced disease”). More generally, any change aimed at reducing the amount of false negatives (such as adding negation triggers) or false positives (such as restricting scope) is likely to induce a commensurate increase in false positives or negatives, respectively. The list of rules—and thereby the trade-off between recall and precision—will have to be adapted and optimized for each individual corpus and application.

### **BiLSTM**

As shown in Table 4, for the biLSTM classifier around 5% of the errors are annotation errors, where the model actually predicted the correct label. For false negatives one of the largest categories is scope errors, which includes examples where a list of entities is negated using a single negation term, as well as examples where many tokens are present between the negation term and the medical term. For false positives, *negation of a different term* is a common error, which is problematic for all three methods. However, compared to the other methods the biLSTM has a more even distribution over error categories. The overall performance and the distribution over categories shows that the biLSTM is more robust against syntactic variation than the rule-based model, but not as generalized as RobBERT.

	Negated	NotNegated		Negated	NotNegated
16 ĠC	0.818836	0.181164	17 Ġoe	0.875645	0.124355
17 oel	0.870450	0.129550	18 deem	0.358316	0.641684
18 iak	0.334840	0.665160			
19 ie	0.034791	0.965209			

(a) Varying negation-estimated over single word *Coeliakie*      (b) Varying negation-estimated over single word *Oedeem*

**Fig. 6** Intra-word negation variance. The token delimiter character Ġ is a result of the tokenization

**RobBERT**

*Negation of different terms, speculation, uncommon negation* and the use of a hyphen (*minus*) to indicate negations are the largest potentially resolvable contributors to the RobBERT error categories, totalling to about 50% of the errors (Table 4).

RobBERT-base fills in missing interpunction, i.e. it expects interpunction based on the corpus it was trained on and in our clinical case we often find that interpunction is missing. A degenerative example is “*The patient is suffering from palpitations, shortness of breath and uedema, this can be an indication of <mask>*”, RobBERT filled in the mask as : (a semicolon).

A large percentage of the false negatives were due to RobBERT mishandling hyphens. We also observed varying model output based on negation triggers being mixed lower/uppercase. Mishandling the hyphens can potentially be resolved by adapting the tokenizer to include the hyphen as a separate token or by adding white space.

We observed that words could have a varying negation estimate over the different tokens that make up the words, illustrated in Fig. 6. This variance is an artefact of using a sub-word tokenizer. This is potentially problematic for words consisting of many tokens, but it also allows for more flexibility because we can decide to (for example) take the maximum probability over the tokens per word. It can also occur that token-specific negations are required, for instance when the negation and the term are concatenated, as in “*De patient is tumorvrij*” (“*The patient is tumorfree*”). The possibility of concatenation is language dependent.

The categories *uncommon negation* and *negation of a different term* can be reduced by expanding the training set with the appropriate samples.

**Discussion**

We compared a rule-based method based on ContextD, a biLSTM model using MedCAT and (finetuned) Dutch RoBERTa-based models on Dutch clinical text and found that both machine learning models consistently outperform the rule-based model in terms of the F1, precision and recall. Combining the three models was not beneficial in terms of performance. The best performing models achieve an F1-score of 0.95. This is a relatively high score for a cross-validated machine learning approach, and is likely near the upper bound of what is achievable for this dataset, considering the noise in labeled data (0.90-0.94 inter-annotator agreement).



### **Applicability**

The performance of the assessed methods is well within the acceptable range for use in many information retrieval and data science use-cases in the healthcare domain. Application of these methods can be especially useful for automated tasks where a small number of errors is permitted, such as reducing the number of false positives during cohort selection for clinical trial recruitment. In this task, erroneously excluding a patient is less problematic, and the included patients can be checked manually for eligibility. Other data applications can benefit from this as well, such as text mining for identification of adverse drug reactions, feature extraction for predictive analytics or evaluation of hospital procedures.

However, the model still makes classification errors, which means it is not suitable as a stand-alone method to retrieve automatic annotations for medical decision support systems but can be used directly to improve existing label extraction processes. Application in a decision support system would require some sort of manual interaction with a specialist.

### **Additional aspects for model comparison**

The model comparison (based on precision, recall and F1 score) shows that the RobBERT-based models result in the highest performance. However, additional considerations can play a role in selecting a model, for example computational and human resources. Fine-tuning and subsequently applying a BERT-based model requires significant hardware and domain expertise, which may not be available in clinical practice, or only available outside the medical institution's domain infrastructure, which introduces security and privacy concerns.

In contrast, the biLSTM and rule-based models can be used on a personal computer, with only a limited performance decrease ( $\sim 0.03$  and  $\sim 0.08$  respectively) on each evaluation metric. The rule-based method has the advantage that model decisions are inherently explainable, by showing the applied rules to the end user. This may lead to faster adoption of such a system compared to black box neural models.

The used biLSTM method is part of MedCAT, which also incorporates named entity recognition and linking methods. Compared to the other assessed approaches, this is a more complete end-to-end solution for medical NLP, and is relatively easy to deploy and use, especially in combination with the information retrieval and data processing functionalities of its parent project CogStack [21]. Recently, MedCAT added support for BERT-based models for identification of contextual properties.

### **Limitations and future work**

The study described in this paper has various limitations for which potential improvements can be identified. Regarding language models, the biLSTM network is trained on a relatively small set of word embeddings obtained from Dutch medical Wikipedia articles. This could be complemented or replaced with a larger and more representative dataset, to be more in line with the language models used in the RobBERT experiments. Alternatively, a corpus of actual electronic health records can be used

for the best representative dataset, yet this leads to privacy concerns given the high concentration of identifiable protected health information in natural language, even after state of the art pseudonimization.

In the current approach the candidate terms for negation detection in the DCC are generated by performing medical named entity recognition, where each recognized entity is presented to the negation detection models. The context provided to each model, which is assumed to be sufficient for determining the presence of negation, is defined to be the sentence around a term as determined by a sentence splitting algorithm. This results in scope-related errors such as incorrect delimitation of the medical term, incorrect sentence splitting, or the negation trigger being in a grammatically different part of the sentence. These issues can be reduced at various stages in the pipeline, either by improving the involved components, or by performing a sanity check on the generated example using part-of-speech-tagging or (dependency) parsing. Another approach to reduce the number of problematic candidates is to train additional classifiers on meta-properties like temporality (*patient doesn't remember previous occurrences of X*) or experiencer (*X is not common in family of patient*). Furthermore, the domain-specific structure of the EHR records in the various categories could be leveraged, e.g., to discard non-relevant sections of the health record during processing for specific use cases.

Considering that several error categories are related to the availability of training data, we can to some extent improve the models using synthetic data or a larger set of manually annotated real data. This can alleviate the lack of balance between the *negation* and *non-negation* classes in the Dutch Clinical Corpus (currently 14% negations), which are problematic for both the biLSTM and the RobBERT models. Furthermore, we observe a significant amount of errors due to, or related to, ambiguity. Such errors are expected, not having errors related to ambiguity could indicate an overfitted model. This idea of error categorisation can also be extended to create a model for estimating the dominant error types in unseen data, i.e. to facilitate model selection and problem-specific model improvements.

In future work it is of interest to train the methods on a broader set of health record corpora, in order to increase the amount of data in general, making it less dependent on DCC specific distributions, and to alleviate the class balance and sparsity issues in particular.

In this work we compared a small number of methods, and this may have led to a conservative estimate on the performance of the resulting ensemble method. For future work it may be interesting to investigate a bespoke ensemble method where rule-based and machine-learning based methods are combined in a complementary fashion. One technique that is particularly interesting is based on *prompting*, which does not require any finetuning and thus allows pre-trained language models to be leveraged directly.

Unraveling the semantics of clinical language in written electronic health records is a complex task for both algorithms and human annotators, as we experienced during error analysis. However, the three assessed methods show a good performance on predicting negations in the Dutch Clinical Corpus, with the machine learning methods producing the best results. Given the sparse availability of NLP solutions for the

Dutch clinical domain, we hope that our findings and provided implementations of the models will facilitate further research and the development of data-driven applications in healthcare.

## Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s12859-022-05130-x>.

**Additional file 1: A.** description of datasets using text examples. **S1.** Comparison of different RobBERT/RobERTa versions, using a batchsize of 32, 3 epochs, a gradient of  $10^{-4}$  and a maximum sequence length of 512. **S2.** Overview of RobBERT/RobERTa results for different batch sizes and token windows. **S3.** Information on the corpora that were used for the Domain-Adapted-Pre-Training

### Acknowledgements

We'd like to express our thanks to Jan Kors from the Biosemantics group at ErasmusMC for providing us with the Dutch Clinical Corpus. Also, we thank UMC Utrecht's Digital Research Environment-team for providing high performance computation resources.

### Author contributions

Conceptualization: BVE; Methodology: BVE, LCR, MS, SCT, MMH, SRSA; Software: BVE, LCR, MS, SCT; Validation: BVE, LCR, MS, SCT; Formal analysis: BVE, LCR, MS, SCT, SRSA; Investigation: BVE, LCR, SCT, MS; Data Curation: BVE, LCR, SCT, MS, SRSA; Writing - Original Draft: BVE, LCR, MS, SCT; Writing - Review & Editing: BVE, LCR, MS, SCT, MMH, SRSA, MARR, SH; Visualization: BVE, SRSA; Supervision: BVE, SH; Funding acquisition: BVE, SH; All authors read and approved the final manuscript.

### Funding

Not applicable

### Availability of data and materials

The Erasmus Dutch Clinical Corpus dataset can be requested from the Erasmus MC <https://biosemantics.erasmusmc.nl/index.php/resources/emc-dutch-clinical-corpus>. Code and analysis notebooks are publicly available on GitHub <https://github.com/umcu/negation-detection> under the MIT license.

## Declarations

### Ethics approval and consent to participate

Not applicable

### Consent for publication

Not applicable

### Competing interests

BvE: None to declare; LCR: None to declare; SCT: None to declare; MS: None to declare; MMH: None to declare; SRSA: None to declare; MARR: None to declare; SH: None to declare;

Received: 20 September 2022 Accepted: 30 December 2022

Published online: 09 January 2023

## References

- Zubair Afzal, et al. ContextD: an algorithm to identify contextual properties of medical terms in a Dutch clinical corpus. *BMC Bioinform.* 2014;15(1):1–12.
- Agarwal S, Yu H. Biomedical negation scope detection with conditional random fields. *J Am Med Inform Assoc.* 2010;17(6):696–701. <https://doi.org/10.1136/jamia.2010.003228>.
- Aronow DB, Fangfang F, Croft WB. Ad hoc classification of radiology reports. *J Am Med Inform Assoc.* 1999;6(5):393–411.
- Bodenreider O. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Res.* 2004;32(suppl 1):D267–70.
- Chapman W, et al. A simple algorithm for identifying negated findings and diseases in discharge summaries. *J Biomed Inform.* 2001;34(5):301–10.
- Chapman W et al. Evaluation of negation phrases in narrative clinical reports. In: Proceedings of the AMIA Symposium. American Medical Informatics Association, vol. 105, 2001b.
- Costumero R et al. An approach to detect negation on medical documents in Spanish. In: International conference on brain informatics and health. Springer; 2014. pp 366–375.
- Cotik V, Roland R, et al. Negation detection in clinical reports written in German. In: Proceedings of the fifth workshop on building and evaluating resources for biomedical text mining (BioTxtM2016). Osaka, Japan: The COLING 2016 Organizing Committee; 2016. pp. 115–124. <https://aclanthology.org/W16-5113>.

9. Cotik V, Stricker V, et al. Syntactic methods for negation detection in radiology reports in Spanish. In: Proceedings of the 15th workshop on biomedical natural language processing, BioNLP 2016: Berlin, Germany, 2016. Association for Computational Linguistics; 2016. pp. 156–165.
10. Cruz Díaz Noa P, et al. A machine-learning approach to negation and speculation detection in clinical texts. *J Am Soc Inf Sci Technol*. 2012;63(7):1398–410.
11. Deléger L, Grouin C. Detecting negation of medical problems in French clinical notes. In: Proceedings of the 2nd ACM sighth international health informatics symposium; 2012. pp. 697–702.
12. Delobelle P, Winters T, Berendt B. RobBERT: a Dutch RoBERTa-based language model. *Find Assoc Comput Linguist EMNLP*. 2020;2020:3255–65.
13. Elazhary H. NegMiner: an automated tool for mining negations from electronic narrative medical documents. *Int J Intell Syst Appl*. 2017;9:14–22. <https://doi.org/10.5815/ijisa.2017.04.02>.
14. Eyre H, et al. Launching into clinical space with medspaCy: a new clinical text processing toolkit in Python. In: Proceedings of the AMIA annual symposium 2021. AMIA. 2021.
15. Gage P. A new algorithm for data compression. *C Users J*. 1994;12(2):23–38.
16. Gkotsis G et al. Don't let notes be misunderstood: a negation detection method for assessing risk of suicide in mental health records. In: Proceedings of the third workshop on computational linguistics and clinical psychology; 2016. pp. 95–105.
17. Goryachev S et al. Implementation and evaluation of four different methods of negation detection. Technical report, DSG: Tech. rep; 2006.
18. Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw*. 2005;18(5–6):602–10.
19. Harkema H et al. ConText: an algorithm for determining negation, experienter, and temporal status from clinical reports. *Journal of Biomedical Informatics* 42.5. *Biomedical Natural Language Processing*, 2009;839–851. issn: 1532-0464. <https://doi.org/10.1016/j.jbi.2009.05.002>. <http://www.sciencedirect.com/science/article/pii/S1532046409000744>.
20. Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF Models for Sequence Tagging. In: CoRR 2015. [arXiv:1508.01991](https://arxiv.org/abs/1508.01991).
21. Jackson R, et al. CogStack-experiences of deploying integrated information retrieval and extraction services in a large National Health Service Foundation Trust hospital. *BMC Med Inf Decis Mak*. 2018;18(1):1–13.
22. Devlin J, Chang M-W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT; 2019. pp. 4171–4186.
23. Khandelwal A, Sawant S. NegBERT: a transfer learning approach for negation detection and scope resolution. In: Proceedings of the 12th language resources and evaluation conference. Marseille, France: European Language Resources Association; 2020. pp. 5739–5748. isbn: 979-10-95546-34-4. <https://www.aclweb.org/anthology/2020.lrec-1.704>.
24. Kim J-D, Ohta T, Tsujii J. Corpus annotation for mining biomedical events from literature. *BMC Bioinform*. 2008;9(1):1–25.
25. Kraljevic Z et al. Multi-domain clinical natural language processing with MedCAT: the medical concept annotation toolkit. In: arXiv preprint 2020. [arXiv:2010.01165](https://arxiv.org/abs/2010.01165).
26. Lample G, Conneau A. Cross-lingual language model pretraining. In: arXiv preprint 2019 [arXiv:1901.07291](https://arxiv.org/abs/1901.07291).
27. Lin C, et al. Does BERT need domain adaptation for clinical negation detection? *J Am Med Inf Assoc*. 2020;27(4):584–91. <https://doi.org/10.1093/jamia/ocaa001>.
28. Liu Y et al. RoBERTa: a robustly optimized BERT pretraining approach. In: ArXiv abs/1907.11692. 2019.
29. Mascio A et al. Comparative analysis of text classification approaches in electronic health records. In: arXiv preprint 2020. [arXiv:2005.06624](https://arxiv.org/abs/2005.06624).
30. McHugh ML. Interrater reliability: the kappa statistic. *Biochem Med*. 2012;22(3):276–82.
31. Mehrabi S, et al. DEEPEN: a negation detection system for clinical text incorporating dependency relation into NegEx. *J Biomed Inform*. 2015;54:213–9. <https://doi.org/10.1016/j.jbi.2015.02.010>.
32. Mukherjee P, et al. NegAIT: a new parser for medical text simplification using morphological, sentential and double negation. *J Biomed Inform*. 2017;69:55–62.
33. Mutalik PG, Deshpande A, Nadkarni PM. Use of general-purpose negation detection to augment concept indexing of medical documents: a quantitative study using the UMLS. *J Am Med Inform Assoc*. 2001;8(6):598–609. <https://doi.org/10.1136/jamia.2001.0080598>.
34. Oostdijk N, et al. The construction of a 500-million-word reference corpus of contemporary written Dutch. In: Essential speech and language technology for Dutch. Berlin, Heidelberg: Springer; 2013. p. 219–47.
35. Peng Y et al. NegBio: a high-performance tool for negation and uncertainty detection in radiology reports. In: AMIA joint summits on translational science proceedings. AMIA Joint Summits on Translational Science 2017. PMC5961822[pmcid], 2018; pp. 188–196. issn: 2153-4063.
36. Pyysalo S, et al. BioInfer: a corpus for information extraction in the biomedical domain. *BMC Bioinform*. 2007;8(1):1–24.
37. Raffel C, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J Mach Learn Res*. 2020;21(140):1–67.
38. Shi J, Hurdle JF. Trie-based rule processing for clinical NLP: A use-case study of n-trie, making the ConText algorithm more efficient and scalable. *J Biomed Inform*. 2018;85:106–13. <https://doi.org/10.1016/j.jbi.2018.08.002>.
39. Slater LT, et al. A fast, accurate, and generalisable heuristic-based negation detection algorithm for clinical text. *Comput Biol Med*. 2021;130:104216.
40. Sohn S, Wu S, Chute CG. Dependency parser-based negation detection in clinical narratives. In: AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science 2012. PMC3392064[pmcid], 2012;1–8. issn: 2153-4063. <https://pubmed.ncbi.nlm.nih.gov/22779038>.
41. Stausberg J, et al. Reliability of diagnoses coding with ICD-10. *Int J Med Inform*. 2008;77(1):50–7.

42. Sun K et al. Aspect-level sentiment analysis via convolution over dependency tree. 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 2019, Hong Kong, China; 2019. pp. 5678–5687.
43. Vaswani A et al. Attention is all you need. In: 2017 [arxiv:1706.03762](https://arxiv.org/abs/1706.03762).
44. Verkijk S, Vossen P. MedRoBERTa. nl: a language model for Dutch electronic health records. *Comput Linguist The Neth J*. 2021;11:141–59.
45. Vincze V. Speculation and negation annotation in natural language texts: what the case of BioScope might (not) reveal. In: Proceedings of the workshop on negation and speculation in natural language processing; 2010. pp. 28–31.
46. Vincze V, et al. The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. *BMC Bioinform*. 2008;9(11):1–9.

### **Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Ready to submit your research? Choose BMC and benefit from:**

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

**At BMC, research is always in progress.**

Learn more [biomedcentral.com/submissions](https://biomedcentral.com/submissions)

