# Not another Negation Benchmark: The NaN-NLI Test Suite for Sub-clausal Negation

**Hung Thinh Truong**[1,*] **Yulia Otmakhova**[1,*] **Timothy Baldwin**[1,3]
**Trevor Cohn** [1] **Jey Han Lau**[1] **Karin Verspoor**[2,1]
[1]The University of Melbourne [2]RMIT University [3]MBZUAI
{hungthinht,yotmakhova}@student.unimelb.edu.au tb@ldwin.net
trevor.cohn@unimelb.edu.au jeyhan.lau@gmail.com karin.verspoor@rmit.edu.au

## Abstract

Negation is poorly captured by current language models, although the extent of this problem is not widely understood. We introduce a natural language inference (NLI) test suite to enable probing the capabilities of NLP methods, with the aim of understanding sub-clausal negation. The test suite contains premise–hypothesis pairs where the premise contains sub-clausal negation and the hypothesis is constructed by making minimal modifications to the premise in order to reflect different possible interpretations. Aside from adopting standard NLI labels, our test suite is systematically constructed under a rigorous linguistic framework. It includes annotation of negation types and constructions grounded in linguistic theory, as well as the operations used to construct hypotheses. This facilitates fine-grained analysis of model performance. We conduct experiments using pre-trained language models to demonstrate that our test suite is more challenging than existing benchmarks focused on negation, and show how our annotation supports a deeper understanding of the current NLI capabilities in terms of negation and quantification.

## 1 Introduction

Negation is an important linguistic phenomenon which denotes non-existence, denial, or contradiction, and is core to language understanding. NLP work on negation has mostly focused on detecting instances of negation (Peng et al., 2018; Khandelwal and Sawant, 2020; Truong et al., 2022), and the effect of negation on downstream or probing tasks (Kassner and Schütze, 2020; Ettinger, 2020; Hossain et al., 2020). A consistent finding in recent work on pre-trained language models (PLMs) is that they struggle to correctly handle negation, but also that existing NLP benchmarks are deficient in terms of their relative occurrence and variability of negation (Barnes et al., 2021; Tang et al., 2021; Hossain et al., 2022).

In this work, we address the problem of evaluating the ability of models to handle negation in the English language using a systematic, linguistically-based approach. Specifically, we adopt the typology proposed by Pullum and Huddleston (2002) whereby negation is classified based on both form (verbal and non-verbal; analytic and synthetic) and meaning (clausal and sub-clausal; ordinary and meta-linguistic). Based on this typology, we observe that most negation instances occurring in existing benchmarks are analytic, verbal, and clausal, which is arguably more straightforward to handle than non-verbal, synthetic, and sub-clausal negation. For instance, the dataset proposed by Hossain et al. (2020) is constructed by adding the syntactic negation cue *not* to the main verb of the premise and/or the hypothesis of MNLI (Williams et al., 2018) training examples, resulting almost exclusively in verbal, analytic, and clausal negation.

Motivated by this, we construct a new evaluation dataset with a focus on sub-clausal negation, where it is non-trivial to determine the correct negation scope. For instance, the negation in *They saw not one but three dolphins* only scopes over the modifier *one*, and thus carries a positive meaning (*They saw three dolphins*). We choose NLI as the probing task based on the intuition that a complete grasp of negation is required to make correct inference judgements. Moreover, we adopt the test suite framework (Lehmann et al., 1996) instead of naturally-occurring text corpora, to elicit a full range of linguistic constructions that denote sub-clausal negation. This facilitates systematic evaluation of model performance along controlled dimensions. We collect examples for each construction from Pullum and Huddleston (2002) to use as premises, and then construct corresponding hypotheses by introducing minimum changes to premises which highlight their possible interpreta-

[*]Equal contribution



*Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 883–894
November 20–23, 2022. 

tions. We manually annotate the constructed pairs in terms of negation types, negation constructions, and the operations used to construct the hypotheses.

In summary, our main contributions are:

1. We introduce the "NaN-NLI" test suite for probing the capabilities of NLP models to capture sub-clausal negation.[1] In addition to standard NLI labels, it contains various linguistic annotations related to negation, to facilitate fine-grained analysis of different constructional and semantic sub-types of negation;

2. We conduct extensive experiments to confirm that our test suite is more difficult than existing negation-focused NLI benchmarks, and show how our annotations can be used to guide error analysis and interpretation of model performance; and

3. We present a subset of our test suite (NaN-Quant) with samples involving not only negation but also quantification, and show that quantification is an especially challenging phenomenon that requires future exploration.

## 2 Related Work

To investigate the abilities of PLMs to assign the correct interpretation to negation, many probing tasks have been proposed. For instance, Kassner and Schütze (2020); Ettinger (2020) formulated a cloze-style fill-in-the-blank task where BERT is asked to predict words for two near-identical but contrasting sentences (e.g. *A bird can* ___ vs. *A bird cannot* ___). Hossain et al. (2020) constructed an NLI dataset where negations essential to correctly judge the label for a premise–hypothesis pair were manually added to existing NLI benchmarks. Hartmann et al. (2021) constructed a multilingual dataset with minimal pairs of NLI examples to analyze model behavior in the presence/absence of negation. Most recently, Hossain et al. (2022) conducted a comprehensive analysis of the effect of negation on a wide range of NLU tasks in the GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019) benchmarks. These papers expose various limitations of both current benchmarks and PLMs in the face of negation. However, they all focus on verbal and clausal negation, which are more straightforward to process, whereas our dataset targets non-verbal and sub-clausal negation, where it is more difficult to determine the correct scope.

The idea of using a test suite to measure the performance of NLP models was introduced by Lehmann et al. (1996), where the authors propose general guidelines for test suite construction. Adopting this methodology for a domain-specific task, Cohen et al. (2010) constructed a dataset for benchmarking ontology concept recognition systems. Most recently, Ribeiro et al. (2020) proposed a task-agnostic testing methodology which closely follows the idea of behavioral testing from software engineering to comprehensively test the linguistic capabilities of NLP models. The main advantages of test suites over datasets made up of naturally-occurring examples are: (1) *control over the precise composition of the data*: we can undertake a targeted evaluation of specific criteria (e.g. linguistic features); (2) *systematicity*: a test suite has specific structure, with samples classified into well-defined categories; and (3) *control of redundancy*: we can remove samples with similar properties or over-sample rare occurrences.

## 3 A Test Suite for Non-verbal Negation

### 3.1 Negation typology

According to Pullum and Huddleston (2002), negation can be classified according to four main aspects:

- **Verbal vs. non-verbal**: verbal negation is when the negation marker is associated with the verb, while non-verbal negation is associated with an adjunct or object.

- **Analytic vs. synthetic**: when the negation marker's only syntactic function is to mark negation (e.g. *not*), it represents analytic negation, whereas in synthetic negation the marker can have other syntactic functions (e.g. a compound negator *nothing* can also be a subject or an object).

- **Clausal vs. sub-clausal**: Clausal negation negates the entire clause it is contained in, whereas the scope of sub-clausal negation is strictly less than the entire clause. For instance, in *Not for the first time, she felt utterly betrayed*, only the phrase *Not for the first time* is negated.

[1] The test suite and all code are available at https://github.com/joey234/nan-nli

- **Ordinary vs. meta-linguistic**: meta-linguistic negation acts as a correction to how the negative meaning is understood. For instance, in *The house is not big, it is huge*, the negation is understood as a correction, since *huge* is a more correct way of describing the size of the house.

The first two categories relate to the syntax of negation itself while the last two relate to semantics. In this work, we focus on sub-clausal negation as the correct negation scope can be challenging to determine, which can lead to misunderstanding of the negated instance. Although meta-linguistic negation can also cause difficulties with interpretation, as this class is rare in practice, we did not include them in our test suite.

### 3.2 Test suite construction process

#### 3.2.1 Selecting premises

We manually collect sentences from Pullum and Huddleston (2002) to use as premises. Most samples are special constructions of non-verbal negation where they denote sub-clausal negation. Below we describe the main types of these constructions.

***Not* + quantifiers**: *not* combines with a quantifier and scopes only over that quantifier.

*Not all*: *not* is used to deny the larger amount, and imply a normal value. Possible quantifiers include *not all, not every, not many, not much, not often*.

*Not one, not two*: *not one* is used to denote a complete non-existence of something, and has the same meaning as *nothing* or *no one*. When combining with a numbers larger than one (usually in phrases of time and distance), *not* can convey the meaning of *as little as*, as in *not two years ago*.

*Not a little*: This construction negates the lower bound of the quantification and asserts the upper bound, denoting *a fairly large amount*. For instance, *not a little confusion* is equivalent to *much confusion*.

***Not* + focus particles (*even/only*):** *Not even* generally marks clausal negation while *not only* marks sub-clausal negation as it carries positive meaning. For instance, *Not even Ed approved of the plan* implies that Ed did not approve the plan, whereas in *Not only Ed approved of the plan*, Ed did in fact approve the plan.

***Not* + degree expressions**: Expressions such as *not very, not quite* mark sub-clausal negation by reducing the degree of adjectives, adverbs, or determiners (e.g. *not very confident*).

***Not* + affixially-negated adjectives/adverbs:** When accompanied by a gradable adjective, the construction *not un-* has the meaning of negating the lower end of the scale for that adjective. For example, *not unattractive* suggests the appearance ranks higher than intermediate.

***Not* in coordination:** *Not* can appear in a coordinative construction and typically scopes over only one of the coordinating parts, thus marking sub-clausal negation. In *They are now leaving not on Friday but on Saturday*, *not* scopes only over *Friday* and denies *They are leaving on Friday.*

***Not* with PPs:** *Not* can modify prepositional phrases (PPs) to denote sub-clausal negation. In *Not for the first time, she felt utterly betrayed*, *not* only negates the PP *for the first time*, and the sentence has positive polarity in that she did feel utterly betrayed.

***Not* in verbless subordinate clauses:** *Not* can scope only over a verbless subordinate clause (e.g. *We need someone not afraid of taking risks.*).

***Not* in implicit propositions with *that***: The construction *not that* has the function of denying something that is natural or expected in the context (e.g. *There are spare blankets in here, not that you'll have any need of them.*).

**Absolute and approximate negators:** Absolute negators (*no, never*) denote absolute non-existence but can also denote sub-clausal negation when they are part of a prepositional phrase. In *They were friends in no time*, only the PP *in no time* is negated. Approximate negators (*rarely, seldom*) denote a quantification that is close to zero. They imply positive meaning and thus denote sub-clausal negation.

#### 3.2.2 Constructing premise–hypothesis pairs

When constructing hypothesis sentences for premises, we aimed to keep lexical changes to a minimum. This was especially so in the case of neutral hypotheses: though it is trivial to create any number of neutral hypotheses by changing semantically important parts of a sentence to other lexical items thus making it impossible to determine the truth value, intuitively, it would make the sentence embedding of the hypothesis quite different from that of the premise and thus easier for models to classify correctly. We also strove to make hypotheses linguistically diverse by introducing various changes to functional words rather than relying only on deletion and addition of *not* as was done

previously. Overall, we used 10 operations, with more than half the hypotheses including two or more changes. They are listed in Table 1 together with representative examples and their frequency counts across all sentences.

As outlined above, when creating hypotheses, we employed a much wider variety of linguistic operations than previous datasets, including movement of a negation marker across constituent boundaries, changing its type or scope, and substitution of indefinite pronouns. Thus we expect our dataset to be both richer and more difficult from the point of view of NLU. On average, for each of the selected premises, we created 5 hypotheses.

#### 3.2.3 Annotating the inference relationship within premise–hypothesis pairs

Following Giampiccolo et al. (2007), we adopt a three-way classification of inference relationships between the premise ($p$) and the hypothesis ($q$) based on the following truth values:

- **Entailment**: if $p$ is True, $q$ must be True.
- **Contradiction**: if $p$ is True, $q$ must be False.
- **Neutral**: if $p$ is True, $q$ can be both True and False, and the available context does not allow us to make a specific judgement.

Two annotators (the main authors of the paper, one of whom holds a graduate degree in linguistics) labeled all constructed pairs with these categories; disagreements were resolved via discussion. The inter-annotator agreement prior to adjudication was 0.86 in terms of Cohen's $\kappa$ (Cohen, 1960), which corresponds to near-perfect agreement (Artstein and Poesio, 2008). We employed the following linguistic tests to distinguish between entailed and neutral pairs (Kroeger, 2018; Anderson, 2018):

- It should be impossible to deny $q$ while asserting $p$, that is, to connect $p$ and $p$ using such expressions as *but it is not the fact that ...*
- It should be unnatural to express doubt about $q$ while asserting $p$, that is, to connect them using such expressions as *but I am not sure whether ...*
- It should be highly redundant to assert $q$ after stating $p$, that is, to connect them with such phrases as *In fact ...*

If $q$ fails at least one of these tests, it is considered to be *neutral* to the premise; we regard a hypothesis to be *entailed* only if it passes all three tests. A *contradiction* was defined to be a statement which is the opposite of what is entailed by a premise. For example, given the premise $p =$ *She didn't promise to help him*, the constructed hypotheses can be annotated in the following way:

- **Entailment**: *She didn't promise him help* (fails all three tests).
- **Contradiction**: *She promised to help him* (direct opposite of $p$).
- **Neutral**: *She promised not to help him* (it can be be denied, asserted, and tentatively asserted).

#### 3.2.4 Annotating premise–hypothesis pairs in terms of negation types, patterns, and introduced changes

Finally, the annotators were asked to annotate each sample with respect to the following:

- **Negation types** in both the premise and hypothesis, as described in Section 3.1 (*verbal* vs. *non-verbal*, *analytic* vs. *synthetic*, *clausal* vs. *sub-clausal*).
- **Negation constructions** in the premises, as described in Section 3.2.1. For some constructions, we also specify their sub-types using their representative expressions as names. For example, for *not* +quantifier, we annotate three sub-types which have distinct meanings: *not many*, *not one*, and *not two*.
- **Operations** used to construct hypotheses, as outlined in Table 1.

The initial inter-annotator agreement scores (Cohen's $\kappa$) were 0.99, 0.88, and 0.83, for negation types, negation constructions, and operations respectively, which is close to near perfect as the categories are well-defined in Pullum and Huddleston (2002). All disagreements were then resolved via discussion. We include such detailed linguistic annotation in the test suite to facilitate error analysis and identifying the most problematic cases.

#### 3.2.5 Test suite statistics and comparison with existing negation benchmarks

The statistics of the resulting dataset — named "NaN-NLI" — in terms of label distribution and the

| Operation type | Example | Count |
|---|---|---|
| Indefinite quantifier change (*many, rarely*) | *She rarely goes out these days.* ⇒ *She never goes out these days.* | 74 |
| Numerical quantifier change (*one, twenty*) | *Not for the first time, she felt utterly betrayed.* ⇒ *She felt utterly betrayed for the second time.* | 27 |
| Negator addition or deletion | *Not even Ed approved of the plan.* ⇒ *Even Ed approved of the plan.* | 130 |
| Negator position change | *He was here not ten minutes ago.* ⇒ *He was not here ten minutes ago.* | 101 |
| Negator token change | *Such mistakes are not common.* ⇒ *Such mistakes are uncommon.* | 6 |
| Clause or sub-clause deletion | *Not often do we see her lose her cool like that.* ⇒ *We do not see her often.* | 36 |
| Comparative quantifier change (*more, less*) | *They had found not one mistake.* ⇒ *They had found less than one mistake.* | 20 |
| Focus particle change (*even, only*) | *Not even Ed approved of the plan.* ⇒ *Not only Ed approved of the plan.* | 16 |
| Lexical change | *We had a not very amicable discussion.* ⇒ *We did not have discussion.* | 13 |
| Syntactic change | *Not an accomplished dancer, he moved rather clumsily.* ⇒ *He moved rather clumsily because he was not an accomplished dancer.* | 4 |

Table 1: Types, examples, and counts of operations used to construct hypotheses

| | | Premise | | | Hypothesis | | | |
|---|---|---|---|---|---|---|---|---|
| | Instances | Verbal/ Non-V | Ana/Syn | Clausal/ Sub-C | Verbal/ Non-V | Ana/Syn | Clausal/ Sub-C | None |
| *C* | 117 (45.3%) | 5.2/ 94.9 | 87.2/ 20.5 | 0.9/ 99.2 | 46.2/ 27.4 | 52.1/ 18.8 | 46.2/ 28.2 | 34.2 |
| *E* | 97 (37.6%) | 0.2/ 99.9 | 84.5/ 20.6 | 5.2/ 94.9 | 53.6/ 20.5 | 60.8/ 11.3 | 52.6/ 21.7 | 30.9 |
| *N* | 44 (17.1%) | 6.8/ 93.2 | 100.0/ 18.2 | 6.8/ 93.2 | 43.2/ 20.5 | 61.4/ 2.3 | 43.2/ 20.5 | 36.4 |
| ALL | 258 | 3.5/96.5 | 88.4/ 20.2 | 3.5/ 96.5 | 48.5/ 23.6 | 57.0/ 13.2 | 48.1/ 24.4 | 33.3 |

Table 2: Distribution of class labels for premises-hypothesis pairs and percentage of each types of negation in premises and hypotheses. *C, E, N* denote Contradiction, Entailment, and Neutral, respectively.

types of negation used in premises and hypotheses is presented in Table 2. Following Hossain et al. (2020), we do not enforce a uniform distribution for the Entailment, Contradiction, and Neutral classes but rather focus on constructing fluent and natural continuations which are as close to the premise as possible. Similarly, when constructing hypotheses, it was impossible to adhere to a particular type of negation or even to preserve it in all cases. Thus, while premises mostly have sub-clausal non-verbal negation expressed by synthetic means, the hypotheses exhibit a wider variety of patterns. It should be noted that though we report the distribution of particular negation patterns as a percentage of sentences, the values for categories do not sum to 100% as some sentences contain more than one instance of negation. Lastly, Table 3 shows the distribution of operations for each of NLI labels. In general, we find the distribution to be quite similar for the most common categories, which allows us to claim that we are not creating major artifacts during annotation.

To estimate the difficulty of our benchmark relative to existing benchmarks, we use BERTScore (Zhang et al., 2019) to compare the average similarity between the premise and hypothesis for the three classes. For comparison, we use a subset of the MNLI dataset (Williams et al., 2018) containing only sentences with negation, as extracted by Hossain et al. (2020) ("MNLI-neg" hereafter), and the MNLI subset of the NegNLI dataset proposed by Hossain et al. (2020) ("NegNLI" hereafter). The average similarity scores are presented in Table 4; for the Contradiction and Neutral classes, in brackets we report the absolute difference over the score for the Entailment class to show how difficult it is to differentiate them. It can be seen that in our test suite, hypotheses are substantially more similar to premises than is the case for other datasets; and it

| Operation type | C | E | N |
|---|---|---|---|
| Indefinite quantifier change | 17 | 21 | 10 |
| Numerical quantifier change | 4 | 4 | 14 |
| Comparative quantifier change | 4 | 4 | 8 |
| Negator addition or deletion | 32 | 27 | 33 |
| Negator position change | 24 | 24 | 22 |
| Negator token change | 1 | 2 | 1 |
| Clause or sub-clause deletion | 8 | 9 | 7 |
| Focus particle change | 6 | 3 | 0 |
| Lexical change | 2 | 3 | 5 |
| Syntactic change | 0 | 2 | 0 |

Table 3: Distribution of operation types in each class (%)

| | MNLI-neg | NegNLI | NaN-NLI |
|---|---|---|---|
| Contradiction | 0.88 (*0.02*) | 0.92 (*0.00*) | **0.96** (*0.00*) |
| Entailment | 0.91 | 0.92 | **0.96** |
| Neutral | 0.89 (*0.01*) | 0.90 (*0.02*) | **0.95** (*0.01*) |

Table 4: Average similarity (in terms of BERTScore) between premises and hypotheses for Entailment, Contradiction and Neutral classes.

| | MNLI-neg | NegNLI | NaN-NLI |
|---|---|---|---|
| Contradiction | 0.917 | 0.718 | 0.664 |
| Entailment | 0.834 | 0.656 | 0.648 |
| Neutral | 0.780 | 0.651 | 0.207 |
| All | 0.862 | 0.676 | 0.580 |

Table 5: Results ($F_1$) of RoBERTa-MNLI on existing negation-focused NLI benchmarks. The lowest result for each row is underlined.

is much harder to separate classes based on lexical similarity alone, with the difference between Entailment and Contradiction classes being negligible, and the difference with Neutral being smaller than for other datasets.

## 4 Experiments

### 4.1 Experimental settings

For evaluation, we consider the three settings of:

- *Standard*: a three-way classification task with three labels: Entailment, Contradiction, and Neutral.
- *Binary*: a binary classification task with two labels: Entailment, and Not Entailment, where we consider all Contradiction and Neutral pairs to be Not Entailment.
- *Strict*: We only consider as correct those samples where all hypotheses for a given premise are assigned the correct label (Entailment, Contradiction, or Neutral).

We report $F_1$-score for the *Standard* and *Binary* settings, and Accuracy for the *Strict* setting. Methods investigated include RoBERTa (Liu et al., 2019) and its CueNB (Truong et al., 2022) variant pre-trained with additional negation data augmentation and a negation cue masking strategy. We fine-tune each model on MNLI (Williams et al., 2018) (denoted "-MNLI"), and the MNLI subset of the NegNLI dataset (Hossain et al., 2020) (denoted "-NegNLI").

### 4.2 Main results

For the first experiment, we measure the performance of a baseline RoBERTa model fine-tuned over MNLI on our test suite, in addition to other existing negation-focused NLI datasets. As shown in Table 5, the results for our evaluation set are substantially lower compared to existing NLI datasets. This shows that our dataset contains many challenging instances of negation. The differences are especially stark for the Neutral class, confirming our intuition that making the sentences in a pair as similar as possible would make them more difficult for the model.

Figure 1 provides the confusion matrices of the baseline RoBERTa-MNLI on existing benchmarks. In NaN-NLI, most errors are from over-predicting Entailment. This again shows that the sentences in our pairs are very similar lexically, and also reconfirms the known bias in MNLI that lexical overlap is a strong cue for entailment (McCoy et al., 2019). On the other hand, for MNLI-neg and NegNLI, the performance for the Contradiction class is the highest. This again reveals a bias in MNLI training data, in that if there is negation in either the premise or hypothesis, the labels are more likely to be Contradiction (Gururangan et al., 2018).

Table 6 reports the detailed results for each class across different evaluation settings. Overall, we observe a common trend in that CueNB outperforms the baseline RoBERTa when fine-tuned on the MNLI dataset. This can be explained by the fact that CueNB is pre-trained using more text containing negations, especially non-verbal and synthetic negations (e.g. *no one, nobody*), resulting in better representations for those negation cues. Fine-tuning on the NegNLI dataset further improves performance, with both RoBERTa-MNLI-NegNLI and CueNB-MNLI-NegNLI having comparable performance but RoBERTa performing better for the Contradiction class while CueNB is more accurate for the Neutral class. For the Strict setting, we observe very low results for all models with RoBERTa-MNLI-NegNLI outperforming its CueNB counterpart by one premise. This underlines the difficulty of our test suite, and shows that current methods struggle with sub-clausal negation.

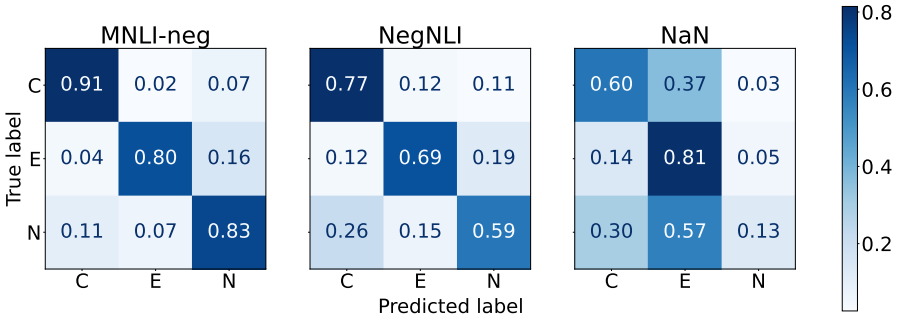


Figure 1: Confusion matrices of RoBERTa-MNLI on different negation-focused NLI benchmarks. *C, E, N* denote the Contradiction, Entailment, and Neutral class respectively.

| | | RoBERTa-MNLI | RoBERTa-MNLI-NegNLI | CueNB-MNLI | CueNB-MNLI-NegNLI |
|---|---|---|---|---|---|
| ***Standard*** | Contradiction | 0.664 | **0.692** | 0.678 | 0.651 |
| | Entailment | 0.648 | 0.684 | 0.678 | **0.694** |
| | Neutral | 0.207 | 0.366 | 0.250 | **0.395** |
| | All | 0.580 | **0.629** | 0.605 | 0.624 |
| ***Binary*** | Entailment | 0.648 | 0.684 | 0.678 | **0.694** |
| | Not Entailment | 0.684 | 0.744 | 0.741 | **0.769** |
| | All | 0.670 | 0.721 | 0.718 | **0.741** |
| | ***Strict*** | 0.250 (12/48) | **0.292 (14/48)** | 0.250 (12/48) | 0.271 (13/48) |

Table 6: Results on our proposed NaN-NLI test suite

## 5 Discussion

We further investigate the results of the best performing model RoBERTa-MNLI-NegNLI in detail to explore potential patterns in the model's predictions on our test suite.

### 5.1 What types of negation are hard?

First, we break down the results by the type of negation used in the premise or hypothesis. There is a substantial difference in performance between samples with analytic and synthetic negation, the latter being more difficult to classify (see Appendix B for details). Considering that in previous datasets negation was expressed primarily by analytic means, we can conclude that the abundance of synthetic negation patterns in our dataset also contributes to its difficulty. In terms of the relation between negation types and inference labels assigned by the models, one significant[2] pattern we notice is that when there is no negation in the hypothesis, models assign Entailment more often. Moreover, there is a significant[2] preference to assign Neutral label when there are analytic negations in the premise compared to synthetic negation. We argue that this is due to the fact that Neutral is the majority class in NegNLI training data.

We further investigate the results based on negation constructions (Section 3.2.1) and operations types (Section 3.2.2). Here, we report error rate, which is the ratio of wrongly predicted samples over all samples in the same construction/modification category. As for linguistic constructions, we find that the most difficult constructions relate to negation in the context of a quantifier, which we further investigate in Section 5.2. Following that, graded adjectives/adverbs, absolute and approximate negators, and degree expressions are among the more challenging construction types for the model to handle. On the other hand, the model deals with coordinations, implicit propositions, and verbless clauses well, with close to zero errors. Following a similar trend, making changes to the quantifiers (either indefinite or comparative) generally confuses the model. We find substantially high error rates for the remaining types of operation except for syntactic change, showing that the model is robust to changing the order of clauses and phrases. Table 7 shows some examples of P-

[2]As determined by the $\chi^2$ test with $p$-value < 0.05

| Premise | Hypothesis | Gold | Predict |
|---|---|---|---|
| Not even then did he lose patience. | Even then, he did not lose patience. | E | E |
| | He did not lose patience even then. | E | E |
| | Not only then did he lose patience. | C | E |
| | Only then did he lose patience. | C | E |
| I found his story not wholly convincing. | I did not find his story wholly convincing. | E | E |
| | I found his story wholly not convincing. | C | E |
| | I found his story wholly convincing. | C | C |
| | I did not find his story wholly not convincing | E | C |
| Not one, not two, but three of them made the mistake. | More than three of them made the mistake. | C | E |
| | More than two of them made the mistake. | E | E |
| | More than one of them made the mistake. | E | E |
| | One of them did not make the mistake. | C | E |
| | Two of them did not make the mistake. | C | N |
| | Less than two of them made the mistake. | C | E |
| | Less than three of them made the mistake. | C | C |
| | Less than four of them made the mistake. | E | E |
| He was here not ten minutes ago. | He was here less than ten minutes ago. | E | E |
| | He was not here less than ten minutes ago. | C | C |
| | He was here more than ten minutes ago. | N | C |
| | He was not here more than ten minutes ago. | N | E |
| | He was not here ten minutes ago. | E | C |
| | He was here one minute ago. | C | N |
| | He was here twenty minutes ago. | N | N |

Table 7: Selected samples along with the predictions of RoBERTa-MNLI-NegNLI. Highlighting is used to indicate prediction errors.

| Construction type | ER |
|---|---|
| *not* + quantifier | **0.559** |
| *not* + focus particle | 0.261 |
| *not* + degree expression | 0.300 |
| *not* + affixially-negated adjective/adverb | 0.423 |
| *not* + PP | 0.067 |
| Absolute and approximate negator | 0.333 |
| *not* in verbless clause | 0.077 |
| *not* in coordination | 0.000 |
| *not* in implicit proposition | 0.000 |

Table 8: Error rates (ER) of negation constructions

| Operation type | ER |
|---|---|
| Indefinite quantifier change | 0.486 |
| Numerical quantifier change | 0.333 |
| Comparative quantifier change | **0.650** |
| Negator addition or deletion | 0.364 |
| Negator position change | 0.327 |
| Negator token change | 0.333 |
| Clause or sub-clause deletion | 0.333 |
| Focus particle change | 0.375 |
| Lexical change | 0.308 |
| Syntactic change | 0.000 |

Table 9: Error rates (ER) across operation types

H pairs, together with their correct and predicted labels.

### 5.2 Using NaN-NLI as a test suite for determining the bounds of quantification

In over half of the samples in our test suite (133), negation interplays with quantification in terms of upper and lower bounds. In the easiest case, if a premise negates a proposition for all members of a set (*None of them supported her*), a contradictory hypothesis would assert that same proposition for any number of members of the set (*One of them supported her*). However, it can be hard even for humans to determine if an expression involving quantification is True or False with regard to another proposition, as it can involve not only indefinite (*any, some, none, many*) and numeric (*one, two, twenty*) quantifiers, but also comparative quantifiers (*more, less*), gradable adjectives (*attractive* → *non unattractive* → *not attractive* → *unattractive*), or adverbs of frequency (*never*, *seldom*, *not often*, *sometimes*, etc). As negation makes this task even harder, we maintain that our test set can be a valuable resource for testing the sensitivity of models to changing of quantification bounds.

As can be seen from Table 10, the performance of the model drops even further on the quantification subset, showing that quantification adds to the difficulty of classification. Interestingly, though, it slightly increases for the Neutral class while plum-

| | NaN-NLI | NaN-Quant |
|---|---|---|
| Contradiction | 0.692 | 0.477 |
| Entailment | 0.684 | 0.600 |
| Neutral | 0.366 | 0.379 |
| All | 0.629 | 0.486 |

Table 10: Results ($F_1$) on the whole NaN-NLI dataset vs. its quantification subset (NaN-Quant). The lowest result for each row is underlined.

meting for the easiest class of Contradiction. We notice that often it is due to inability of the model to detect the lower or upper bound of proposition, that is, where it ceases to hold. For example, here the model correctly predicts Entailment as *more than two* is still within the quantification bounds:

> *Not one, not two, but three of them made the mistake. ⇒ More than two of them made the mistake.*

However, when we increment the number past the bound of *two*, the hypothesis becomes contradictory, but the model fails to detect that and still predicts Entailment, possibly because *three* is also present in the premise:

> *Not one, not two, but three of them made the mistake. ⇒ More than three of them made the mistake.*

In a similar way, such phrases as *not two years ago* implicate a lower bound of the proposition, implying that it is False for numbers smaller than *two*, but the model's prediction of Neutral instead of Contradiction does not reflect that:

> *Not two years ago this company was ranked in the top ten. ⇒ One year ago this company was ranked in the top ten.*

### 5.3 Does gender affect negation?

We manually augment the test suite with simple heuristics to investigate whether gender has an effect on negation. In particular, when the sentences pairs contain a gender-specific pronouns or names, we would generate an equivalent set of sentences pairs with alternate gender pronouns or names (e.g. *he* → *she*, *Ed* → *Sally*). In general, we notice no difference between the original and the gender-altered samples, showing that gender bias does not affect the types of negations in our test suite.

### 5.4 Limitations

The most prominent limitation of our test suite is unbalanced classes distribution, especially for the Neutral class. As discussed in Section 3.2.2, the fact that we try to construct the hypotheses by making minimum edits to the premises would make it very hard to construct meaningful *Neutral* samples. However, we argue that this is acceptable for the evaluation set, as it does not cause bias in training models.

Additionally, our test suite samples are mostly in the general English domain. As shown in previous work (Khandelwal and Sawant, 2020; Truong et al., 2022), the ways that negation is represented varies substantially across domains, and there may be other potentially challenging patterns of negation in other domains or in specific text types (e.g. in clinical notes), as well as other languages (Jiménez-Zafra et al., 2021). These directions we leave for subsequent work.

## 6 Conclusion

In this work, we proposed a new test suite, dubbed NaN-NLI, for probing the performance of NLP models on data containing sub-clausal negation. In addition to standard NLI labels, we also annotated the test suite using a systematic linguistic framework. NaN-NLI facilitates extensive analysis of negation instances based on their negation and construction type. Extensive experiments show that our test suite is significantly harder for existing models than existing benchmarks, and reveal the limited capabilities of pretrained language models in dealing with this type of negation. Detailed analysis of the results reveals a class of negations that are particularly challenging, namely those involving quantifiers, showing that our test suite can also be used as a resource to evaluate the upper and lower bounds of quantification.

## Acknowledgement

The authors would like to thank the anonymous reviewers for their constructive reviews. This research was undertaken using the LIEF HPC-GPGPU Facility hosted at the University of Melbourne. This Facility was established with the assistance of LIEF Grant LE170100200. This research was conducted by the Australian Research Council Training Centre in Cognitive Computing for Medical Technologies (project number ICI70200030) and funded by the Australian Government.

## References


Catherine Anderson. 2018. *Essentials of Linguistics*. McMaster University.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational linguistics*, 34(4):555–596.

Jeremy Barnes, Erik Velldal, and Lilja Øvrelid. 2021. Improving sentiment analysis with multi-task learning of negation. *Natural Language Engineering*, 27(2):249–269.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

K. Bretonnel Cohen, Christophe Roeder, William A. Baumgartner Jr., Lawrence E. Hunter, and Karin Verspoor. 2010. Test suite design for biomedical ontology concept recognition systems. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Mareike Hartmann, Miryam de Lhoneux, Daniel Hershcovich, Yova Kementchedjhieva, Lukas Nielsen, Chen Qiu, and Anders Søgaard. 2021. A multilingual benchmark for probing negation-awareness with minimal pairs. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 244–257, Online. Association for Computational Linguistics.

Md Mosharaf Hossain, Dhivya Chinnappa, and Eduardo Blanco. 2022. An analysis of negation in natural language understanding corpora. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 716–723, Dublin, Ireland. Association for Computational Linguistics.

Md Mosharaf Hossain, Venelin Kovatchev, Pranoy Dutta, Tiffany Kao, Elizabeth Wei, and Eduardo Blanco. 2020. An analysis of natural language inference benchmarks through the lens of negation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9106–9118, Online. Association for Computational Linguistics.

Salud María Jiménez-Zafra, Noa P Cruz-Díaz, Maite Taboada, and María Teresa Martín-Valdivia. 2021. Negation detection for sentiment analysis: A case study in spanish. *Natural Language Engineering*, 27(2):225–248.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7811–7818, Online. Association for Computational Linguistics.

Aditya Khandelwal and Suraj Sawant. 2020. NegBERT: A transfer learning approach for negation detection and scope resolution. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5739–5748, Marseille, France. European Language Resources Association.

Paul Kroeger. 2018. *Analyzing Meaning: An Introduction to Semantics and Pragmatics*. Language Science Press.

Sabine Lehmann, Stephan Oepen, Sylvie Regnier-Prost, Klaus Netter, Veronika Lux, Judith Klein, Kirsten Falkedal, Frederik Fouvry, Dominique Estival, Eva Dauphin, Herve Compagnion, Judith Baur, Lorna Balkan, and Doug Arnold. 1996. TSNLP - test suites for natural language processing. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Yifan Peng, Xiaosong Wang, Le Lu, Mohammadhadi Bagheri, Ronald Summers, and Zhiyong Lu. 2018. NegBio: a high-performance tool for negation and uncertainty detection in radiology reports. *AMIA Summits on Translational Science Proceedings*, 2018:188.

Geoffrey K. Pullum and Rodney Huddleston. 2002. *Negation*, chapter 9. Cambridge University Press.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Gongbo Tang, Philipp Rönchen, Rico Sennrich, and Joakim Nivre. 2021. Revisiting negation in neural machine translation. *Transactions of the Association for Computational Linguistics*, 9:740–755.

Thinh Truong, Timothy Baldwin, Trevor Cohn, and Karin Verspoor. 2022. Improving negation detection with negation-focused pre-training. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4188–4193, Seattle, United States. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

## A Implementation Details

All models are implemented using the `transformers` package from HuggingFace (Wolf et al., 2020). We use the base variant of `RoBERTa`. For fine-tuning on NegNLI, we split the dataset into training/validation sets with a 85:15 ratio.

| Hyper-parameter | Value |
|---|---|
| `batch size` | 16 |
| `lr` | 3e-5 |
| `epochs` | 3 |
| `optimizer` | Adam |

Table 11: Hyper-parameters for fine-tuning on MNLI

| Hyper-parameter | Value |
|---|---|
| `batch size` | 16 |
| `lr` | 2e-5 |
| `epochs` | 5 |
| `optimizer` | Adam |

Table 12: Hyper-parameters for fine-tuning on NegNLI

## B Results by Negation Types

In Table 13 we show the performance of one of the models (`RoBERTa`-MNLI-NegNLI) for samples with a particular type of negation used in the premise or hypothesis. It should be noted that since in the premises negation was almost exclusively non-verbal and sub-clausal, the results for some categories (*Premise - Verbal*, *Premise - Clausal*) are not meaningful.

## C Prediction Examples

| | Negation type | Precision | Recall | $F_1$ |
|---|---|---|---|---|
| ***Premise*** | *Verbal* | 0.39 | 0.60 | 0.46 |
| | *Non-Verbal* | 0.62 | 0.59 | 0.59 |
| | *Analytic* | 0.61 | 0.59 | 0.59 |
| | *Synthetic* | 0.43 | 0.49 | 0.45 |
| | *Clausal* | 0.39 | 0.60 | 0.46 |
| | *Sub-clausal* | 0.62 | 0.59 | 0.59 |
| ***Hypothesis*** | *Verbal* | 0.65 | 0.57 | 0.58 |
| | *Non-Verbal* | 0.63 | 0.59 | 0.57 |
| | *Analytic* | 0.68 | 0.60 | 0.60 |
| | *Synthetic* | 0.48 | 0.45 | 0.41 |
| | *Clausal* | 0.65 | 0.57 | 0.57 |
| | *Sub-clausal* | 0.63 | 0.59 | 0.57 |
| | *None* | 0.60 | 0.57 | 0.58 |

Table 13: Macro-averaged results for RoBERTa-MNLI-NegNLI by negation type