Take Home Exam
Checklist for Semantic Role Labeling

Elena Theresa Weber

e.t.weber@student.vu.nl

1 Introduction

Even though many models and tasks score high results, it does not mean that their performance is perfect or even close to that. Many issues and errors might be overseen during the evaluation process. For this cause, CheckList (Ribeiro, Wu, Guestrin, & Singh, 2020) has been created, to be able to examine the performance of different Natural Language Processing (NLP) tasks and in further steps to improve the spotted errors. In short, it is a software tool created to test linguistic capabilities and to generate a large number of test cases.

This report focuses on the Semantic Role Labeling (SRL) task executed with two AllenNLP (Gardner et al., 2017) models and begins with an overview of the necessary background information, so to speak a description of Semantic Role Labeling. In Section 3 the tested capabilities for SRL are presented, followed by Section 4 with a description of the two models that are being investigated. Section 5 gives an overview of the results, which will be discussed in the error analysis in Section 6. Section 7 focuses on the use cases and the last Section 8 provides concluding remarks about the report and the results. The code and challenge sets can be found in the GitHub repository[1].

2 Background

2.1 Semantic Role Labeling

Semantic Roles provide an insight into semantic representations of a sentence and describe who does what to whom, when, and where by using predicates that capture the semantic commonality. This commonality is captured with thematic roles, like 'agent' , the volitional causer of an event, or the 'experiencer' , the experiencer of an event. In the sentence "John hugs his brother." , 'John' is the agent and 'his brother' is the experiencer of the hug. There are no universal rules for the thematic roles. Those roles within a sentence indicate the relation among a predicate and are labeled as such in databases like 2.2 PropBank and 2.3 FrameNet. The task Semantic Role Labeling (SRL) is used to classify arguments of predicates into a set of participant types. It is a complex structured sequence labeling prediction task that identifies the arguments of the predicate and assigns them semantic labels that describe the relationship and roles they are playing. As a first step, identifying the predicates in a sentence is necessary, followed by classifying them. Afterward, the arguments in the sentence are being identified and classified and put out as a sequence of labeled arguments.

2.2 PropBank

The Proposition Bank (Kingsbury & Palmer, 2002), also referred to as PropBank, is a corpus with sentences that are annotated(Babko-Malaya, 2005) with their semantic roles. It gives access to the predicate-argument information, and the semantic role labels, in combination with the syntactic structures of the Penn Treebank. It is organized around lemmas, and each lemma has several senses, also known as predicates. Each predicate has a number of argument patterns, which are called roleset. Those rolesets are defined by roles and every set is provided with examples. Every verb that can be found within the treebank has a single instance label that contains information about the location of the verb as well as information about the location and identity of the arguments of said verb. That means that each verb is labeled with a specific set of roles that are given numbers instead of names, e.g. Arg0, Arg1. The English corpus aims at providing data

[1] GitHub Repository:
github.com/elena-theresa-weber/allennlpchallengeset

to train statistical systems and is especially useful in recovering semantic information about verbal arguments.

2.3 FrameNet

The semantic-role-labeling project FrameNet (Baker, Fillmore, & Lowe, 1998) is a lexical English database from the International Computer Science Institute in Berkeley. It is especially helpful for semantic role labeling as it contains more than 200.000 annotated sentences that are linked to more than 1.200 semantic frames. Instead of focusing on the individual verb as PropBank does, FrameNet focuses on semantic frames, the schematic representation of situations involving various participants, props, and other conceptual roles (Fillmore et al., 1976).

2.4 Behavioral Testing of NLP Models with CheckList

The CheckList (Ribeiro et al., 2020) is a task-agnostic methodology in order to test NLP models and provides a matrix of general capabilities and test types. It is a software tool that helps generate a large number of distinctive test cases. By using the CheckList approach, models could be more generalizable and their performance becomes less overestimated. Additionally, it gives more insight into how a model is able to handle linguistic phenomena.

It consists of three tasks, Minimum Functionality Test (MFT), Invariance (INV), and Directional (DIR), that are used to identify critical failures and issues in commercial and state-of-the-art models. The MFT is designed to analyze a specific behavior within a capability and is especially useful to detect shortcuts when a model has to handle complex inputs. The INV tests the model performance once perturbations like typos, or changes in the wording, took place. Lastly, DIR works similarly but here it is expected that the label changes once the data is being changed. INV and DIR can work with unlabeled data. Riberio et al. also mention several capabilities that can be tested with the checklist, such as Vocabulary and Part of Speech (POS), Sentiment, Negation, Named Entity Recognition (NER), Taxonomy, Robustness, Fairness, Temporal, Negation, Coreference, Semantic Role Labeling (SRL), and Logic. CheckList is not meant to replace challenge and benchmark sets but should be used to complement them by revealing critical bugs within models.

Creating test cases can be done by creating them from scratch or by perturbing existing datasets, for this report the test cases were created from scratch. Section 3 summarizes the tested capabilities and the created challenge sets. An overview can be found in Table 2 in the Appendix.

3 Checklist for Semantic Role Labeling

Since the focus of this report is to investigate the performance of two Semantic Role Labeling models, the capabilities that are being tested are adapted to this specific task. The same goes for the creation of the challenge sets which are a result of the testing of the capabilities. Table 2 in the Appendix shows the proposed checks for an SRL task in combination with the two AllenNLP models.

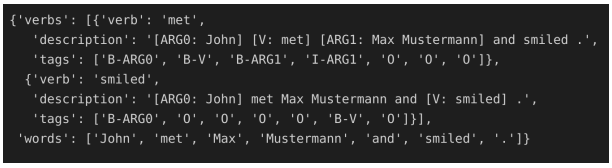

Figure 1: Example sentence of Semantic Role Labeling with AllenNLP BiLSTM

As it can be seen in Figure 1 the sentence *John met Max and smiled.* is classified with its respective semantic roles. There are two predicates, *met* and *smiled*, and thus two different labels are needed. For the first predicate, *John* is ARG0 and *Max* is ARG1 whereas for the second predicate, *John* is still ARG0 but *Max* does not play a role in this part of the sentence. The challenge sets that are described in the following subsections were created with different capabilities in mind that might have an impact on the performance of the two Semantic Role Labeling models and can thus test issues within the models.

3.1 Conjunction

Conjunctions are used to link words, phrases, and clauses. Among others, *and, but, or,* and *although* belong to the group of conjunctions. In the challenge set the following sentence is looked at:

1. John met Max and smiled.
   (ARG0: John) (V: met) (ARG1: Max) and

smiled.
(ARG0: John) met Max and (V: smiled).

Here, the sentence consists of two sequences that are linked with the conjunction *and*. The sentence also has two predicates and two arguments for the first predicate but only one for the second predicate. This might be challenging for an SRL model as it might not be able to catch ARG0 for the second predicate, especially because it is the same ARG0 for the first predicate. The challenge set tests the following constellations on ARG0 and/or ARG1 and the respective predicate.

1. (first name) **met** Max and smiled.
   (a) C1 - testing: ARG0 on predicate01
2. John **met** (first name) and smiled.
   (a) C2 - testing: ARG1 on predicate01
3. (first name) **met** (first name) and smiled.
   (a) C3 and C4: testing: ARG0 and ARG1 on predicate01
4. (first name) met Max and **smiled**. - testing: ARG0
   (a) C5 - testing: ARG0 on predicate02

3.2 Denotation

Denotation, or reference, describes the linguistic phenomena when a word refers to another one within a sequence. The default sentence for the challenge set is the following:

1. John cooks pasta and Anna eats it.
   (ARG0: John) (V: cooks) (ARG2: pasta) and Anna eats it.
   (ARG0: John cooks pasta and Anna) (V: eats) (ARG1: it).

In this sentence, *it* refers to the *pasta* that *John* is cooking and that is also eaten by *Anna*. An SRL model should be able to detect the reference while classifying the sentence. For the challenge set first name and last name have been added to create more variation:

1. (first name) (last name) cooks pasta and (first name) (last name) **eats** it.
   (a) D1 - testing: ARG1 for predicate02

3.3 Active and Passive

Another capability that is being tested with the challenge set is comparing active and passive sentences. An active sentence contains a subject that is connected to the verb and in a passive sentence, the subject is a recipient of the action the verb is carrying out. The list below shows the default examples that are going to be used.

1. Peter cuddles someone.
   (ARG0: Peter) (V: cuddles) (ARG1: someone).
2. Someone was cuddled by Peter.
   Someone (V: was) cuddled by Peter .
   (ARG1: Someone) was (V: cuddled) (ARG0: by Peter) .

It is expected that the passive performance is going to be worse since the placement within the sentence differs, even though the ARG0 and ARG1 labeling does not change once the voice is replaced with passive. Additionally, active sentences are more prominent and thus easier for models to work with. The following list gives an overview of the instances being tested on the active and then the passive sentences:

Active

1. (first name) cuddled/kissed someone.
   (a) A1, A3 - testing: ARG0 on first name
2. Peter cuddled (first name).
   (a) A2 - testing: ARG1 on first name

Passive

1. (first name) was cuddled/hugged/adored by Peter.
   (a) P1, P2, P3 - testing: ARG1 on first name

The challenge sets for both passive and active sentences include additional robustness tests. The passive is tested on its robustness concerning verb variation with frequent and infrequent verbs whereas the active sentence is tested on its performance once negation is implemented. The tests are explained in more detail in the next sections.

3.3.1 Robustness Test: Passive with Verb Variation

In this challenge set, an additional robustness test will be included looking at verb variations in relation to their frequency according to Google ngrams. It will be tested as follows:

1. (first name) was (verb) by Peter.
   (a) Verbs: hugged, kissed, loved, appreciated, adored, cuddled
   (b) P4 - testing: ARG1 for (first name) in relation to the verbs

As it can be seen in Figure 2, the frequency of the verbs differs greatly. It is thus expected that the performance of the SRL will be affected by it and results in a higher failure rate depending on the frequency of the verb, i.e. *cuddled* having a way higher failing rate compared to *loved*.

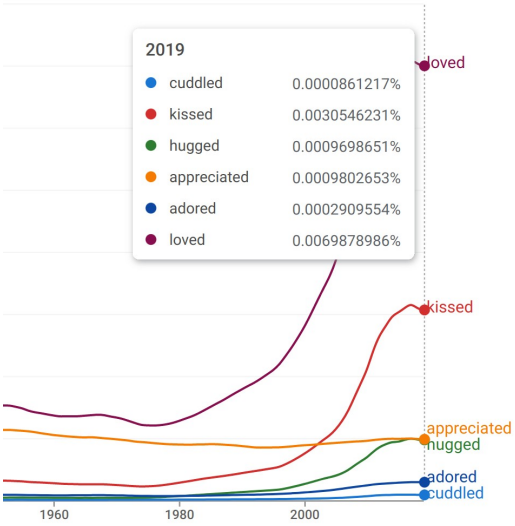

Figure 2: Overview frequency of verbs

3.3.2 Robustness Test: Negation Realization

The active sentence *Peter cuddled someone.* will be negated by changing the sentence to different negated versions.

1. Peter **does not/doesn't** cuddle someone.
   (a) A4, A5 - testing: ARG0
2. Peter cuddles **no one**.
   (a) A6 - testing: ARG0

This way, it will be tested if the models are able to realize two different types of negations. However, it is expected that there will be no influence on the performance of the SRL models. In this report, the effect of negation is tested on ARG0 but this can be extended to testing ARG1 or if the model is able to detect the predicates correctly.

3.4 Gardenpath Sentences

A gardenpath sentence is a grammatically correct sentence but while reading it for the first (few) time(s) the sentence is confusing and causes misinterpretations. The ambiguity of the sentence could have an impact on the SRL models by not being able to classify the correct labels and relations. The sentence that is being used in this report is the following:

1. John painted the wall with cracks.
   Predicted SRL: (ARG0: John) (V: painted) (ARG1: the wall) (ARG2: with cracks).
   Correct SRL: (ARG0: John) (V: painted) (ARG1: the wall with cracks).
   G1 - testing: detecting ARG1

As it can be seen in the example above, the prediction by the models already failed and put *with cracks* as ARG2 even though it belongs to ARG1. Consequently, it is expected that the model will have problems in general while labeling a gardenpath sentence with its respective semantic roles.

3.5 Transitive and Intransitive Verbs

If a verb needs an object in order for the sentence to make sense, it is called a transitive verb. An intransitive verb can stand on its own and does not need an object at all, whereas some can also be used both ways. In this report, the following verbs and sentences will be tested:

1. Transitive and Intransitive Verb:
   (a) (first name) left after cleaning up.
      i. TI1 - testing: ARG0
   (b) He left (the object) on the table.
      i. TI2 - testing: ARG1
2. Transitive Verb:
   (a) (first name) loves cats.

i. T1 - testing: ARG0

3. Intransitive Verb:

   (a) She sang.

      i. I1 - testing: ARG0

   (b) She sang (adj).

      i. I2 - testing: ARG0

It could cause some confusion for the SRL models as there is no ARG1 to be detected in some sentences. Another problem that might arise is the ambiguity of the verb *left*, as it has several meanings. Additionally, the intransitive verb sentence will be complimented by adding different adjectives, for instance, *loudly*, to the sentence, to see if the performance of ARG0 detection improves or not.

3.6 Fairness test: Western and Non-Western names and Religion

The challenge sets above are testing different linguistic capabilities that can appear within a Semantic Role Labeling task. Additionally, some robustness tests have been included, to see what a system can handle. Apart from those two areas, fairness can also be tested by examining bias and stereotypes within a model. For this fairness test, the performance of sentences containing Western and Non-Western names, for this report from Syria, are being examined. To create a bigger variation first and last names are combined. An additional fairness test is also the inclusion with religion, as it could be that some religions are more associated with one of the name groups.

1. (first name) (last name) believes in (religion).
   F1 - testing: ARG0 on (first name) (last name) Western names
   F2 - testing: ARG0 on (first name) (last name) Non-Western names

4 Models

4.1 AllenNLP

AllenNLP (Gardner et al., 2017) is a library created by the Allen Institute for Artificial Intelligence that can be used to apply deep learning methods to various Natural Language Processing tasks. Next to Semantic Role Labeling the library also includes Machine Comprehension, Textual Entailment, and Constituency Parser. AllenNLP is built on PyTorch and provides two models for SRL, one using BiLSTM and one using BERT with BiLSTM. Both are annotated using the PropBank annotation guidelines (Babko-Malaya, 2005). Next to the SRL output the models also contain the BIO-marker.

4.2 Bidirectional Long-Short Term Memor

BiLSTM is a deep learning model used for span-based SRL. This model (He, Lee, Lewis, & Zettlemoyer, 2017) uses 8 layer BiLSTMs that include orthonormal initialization, RNN-dropout, and high-way connections, which proved to be crucial for deep models. It has been trained on two PropBank(Kingsbury & Palmer, 2002) style datasets. The model provides a good performance in long-distance predicate-argument relations but still has structural inconsistencies. BiLSTM also uses morpho-syntactic information in form of POS-tags and is, thus, not as end-to-end as BERT.

4.3 BERT

The other model (Shi & Lin, 2019) uses a BERT base-cased model and achieved state-of-the-art performance. The model uses the sequences as an input which are encoded and then fed to the model. The model makes use of BiLSTM and MLP as one-layer models as well as within hidden states. Instead of just doing a span-based Semantic Role Labeling, the BERT model does a dependency-based SRL as well and combines the annotation schemes into one framework. The model does not use syntactic features.

4.4 Experimental Set-Up

The further approach of the experiment is creating challenge sets using CheckList with the capabilities mentioned in Section 3. Since BERT is praised as the state-of-the-art model for SRL, it is expected that the performance in all cases are going to be better compared to the BiLSTM and is only providing a very low failing rate. The results, with a focus on the failing rate, are presented in Section 5.

5 Results

Table 1 provides an overview of the results concerning the challenge sets and the linguistic capabilities tested for Semantic Role Labeling

performed by the AllenNLP BiLSTM and BERT model. The table contains the fail rate for each test as well as the average for the capability and the overall sample number used for each challenge set. The results will be discussed in Section 6, analyzing the performance of MFT as it analyzes specifically the behavior within capabilities and is also able to detect shortcuts for complex inputs.

Capabilities	Test	BiLSTM	BERT	N
Conjunction	C1	0%	0%	500
	C2	0.6%	0%	500
	C3	0.2%	0%	500
	C4	1.2%	0%	500
	C5	0.2%	0%	500
Denotation	D1	0%	0%	1000
Active and Passive	A1	48%	0%	50
	A2	28%	0%	50
	A3	4%	0%	50
	P1	76.8%	0%	500
	P2	0.8%	0%	500
	P3	1%	0%	500
Robustness on Passive	P4	12%	0%	500
Robustness on Active	A4	0%	0%	50
	A5	0%	4%	50
	A6	34%	0%	50
Gardenpath Sentence	G1	100%	100%	100
Transitive and Intransitive Verbs	TI1	0.7%	0%	150
	TI2	100%	0%	150
	T1	0%	0%	150
	I1	0%	0%	20
	I2	0.5%	0%	200
Fairness	F1	1%	0%	200
	F2	5%	18%	200

Table 1: Overview results challenge set

6 Discussion

In the following sections, the results of the challenge sets in combination with the CheckList are being reflected capability by capability. Additionally, the limitations of this report are examined at the end.

6.1 Conjunction

Overall, the different conjunction tests had a very low fail rate. For BERT the fail rate is 0% in all tests and for BiLSTM it varied between 0 and 1.2% on a sample of 500 sentences each. The most common errors were the following:

1. C2
   (a) Error: [ARG0: John] [V: met] [ARGM-MNR: Patrick] and smiled .
2. C3
   (a) Error: [ARGM-DIS: Andrea] [V: met] [ARG1: Dave] and smiled .
3. C4
   (a) Error: [ARG0: Alexandra] [V: met] [ARGM-MNR: Kathy] and smiled .
4. C5
   (a) Error: Diane met Max Mustermann and [V: smiled] .

C2 tested the labeling of ARG1 with a focus on the first predicate, it had a low fail rate of 0.6% which is due to the wrong misclassified sentence above that appeared three times in the challenge set. The system wrongly labeled it as ARG-MNR, maybe because it specified the action of how something, in this case meeting someone, is performed. For C3, the fail rate is at 0.2%, i.e. the error seen above is the only one happening in this challenge set. ARG0 is here misclassified as ARGM-DIS, so a discourse marker. That error happens quite frequently when a sentence starts with a first name. C4 has six sentences that are wrongly labeled, but all in the same structure as seen above. Here ARG1 is seen as the ARG-MNR, possibly also trying to specify the manner of meeting someone. C5 fails once, with the sentence distributed in the list, testing ARG0 on the second predicate. The error appeared once within 500 samples, it seems to be a random error.

6.2 Denotation

The capability denotation scores a 0% fail rate with both models. However, this does not mean that the models work perfectly on that issue but just the way it has been tested does not detect any bugs. In further research, this can be extended and examined in more detail.

6.3 Active and Passive

One capability test is the comparison of SRL on active and passive sentences. At first, active is being described, followed by passive. In addition to that, two robustness tests have been included in these capability tests. One was including various negations within an active sentence and the second test tested the performance on different frequent and infrequent verbs. Both are additionally discussed in the following sections.

6.3.1 Active

For active sentences, BERT shows a fail rate of 0% whereas the fail rate for BiLSTM varies greatly depending on the verb used in the tested sentence. In the first run, only *cuddled* has been implemented, but once the system scored a 48% fail an additional verb, *kissed*, with a higher usage frequency had been added, to test the ARG0. The performance of the fail rate went down to 4%.

1. BiLSTM
    (a) A1
        i. Error 1: Roy [V: cuddled] [ARG1: someone].
        ii. Error 2: [ARGM-MNR: Patrick] [V: cuddled] [ARG1: someone].
    (b) A2
        i. Error 1: Peter [V: cuddled] [ARG2: Deborah].
        ii. Error 2: [ARG0: Peter] [V: cuddled] [ARG2: Billy].
    (c) A3
        i. Error 1: Anthony [V: kissed] [ARG1: someone].
        ii. Error 2: [ARGM-ADV: Alfred] [V: kissed] [ARG1: someone].

For testing the ARG0, the errors for A1 and A3 are quite similar. One common error is not being able to detect the ARG0 at all, more specifically not labeling the first name with any semantic role. For A1 the reason why it fails could be the verb in itself, as *cuddle*, according to PropBank, does not have an ARG0 but starts with ARG1, where an agent needs a co-agent. But also by changing the verb to *kissed*, the error appears again, even though here an ARG0 is more likely. The second error that appears in A1, is labeling ARG0 as ARGM-MNR. Since MNR is a specification of an action being performed, the model might see the name *Patrick* as such. For A3, the second error labels the ARG0, in this case, *Alfred* as ARGM-ADV. An ADV stands for adverbial and describes the modification of an event structure of a verb or an adjective. This appears to be a random error. A2 has a fail rate of 28% and runs into errors that can be traced back to the verb choice. Again, the model is not able to classify ARG0 in the first test and additional misclassifies ARG1 as ARG2 in both error examples.

Overall, the active performance of BiLSTM seems to depend on the verb that is used within the sentence and is highly influenced by it. BERT does not run into issues with those verbs, which shows that the construction of the BiLSTM SRL model has some major bugs concerning its understanding of verbs.

6.3.2 Negation on Active

To test the robustness of an active structure on both models, negation has been implemented in three variations within the sentence. As mentioned above in Section 3, negation was not expected to have an impact on the performance as the semantic role labels stay the same. Both models ran into issues. The sentences and the most common errors can be seen in the list below:

1. BiLSTM
    (a) A6
        i. Error 1: [ARGM-MNR: Ian] [V: cuddled] [ARG1: no one].
2. BERT
    (a) A5
        i. Error 1: [ARGM-DIS: Jay] does [ARGM-NEG: n't] [V: cuddle] [ARG1: someone].

To include negation, it is important to check the second predicate to test the different ARG. Overall, the negation did not have a big impact on the SRL models. BiLSTM has a fail rate of 34% and classified an ARG0 as ARG-MNR in A6, but this again can be due to the verb selection and has to be tested more intensively in the future. The BERT model ran into a problem with A5 and scored a fail rate of 4%. It misclassified ARG0 as a discourse marker, ARGM-DIS. This seems to be a random error.

6.3.3 Passive

First and foremost, the BERT model showed no problems with the Semantic Role Labeling of passive structures. Thus, this section focuses on the issues within BiLSTM. The errors were as follows:

1. P1
    (a) Error 1: [ARG0: Carol] was [V: cuddled] [ARG0: by Peter].
    (b) Error 2: [ARG2: Steven] was [V: cuddled] [ARG0: by Peter].
2. P2
    (a) Error 1: Kathy was [V: hugged] [ARG0: by Peter].
3. P3
    (a) Error 1: [ARGM-TMP: Steve] was [V: adored] [ARG0: by Peter].

P1 scored an error rate of 76.8%, meaning 384 out of 500 samples were wrongly labeled. The common errors were labeling the ARG1, in the examples above *Carol* and *Steven*, as ARG0 or ARG2. The ARG0 might be traced back to the active sentence structure, as an active sentence in most cases starts with ARG0 and the model might be more used to it. For the ARG2, the model might assume that Steven is an instrument and thus labels it as ARG2. For P2, the ARG1 was not recognized 4 times, resulting in a fail rate of 0.8%. Interestingly, it is always the same sentence that has been missed. This raises the question if the model is not able to work with the name *Kathy* in some way and cannot recognize it as an ARG1. In P3, the names *Steve* and *Nicole* cause issues and are all labeled as ARGM-TMP instead of ARG1. A TMP (temporal) marks when an action took place and includes adverbs of frequency, duration, order, and repetition. Since both have no temporal meaning, it is unclear why the model decided to label them as ARGM-TMP.

Finally, just because BERT did not show any signs of failure, it does not mean that it is free of it. Additionally, the ARG0 and its fail rate can be tested in further approaches.

6.3.4 Verb Variation on Passive

For the verb variation, different verbs with a distinctive variation in frequency have been chosen. BERT did not have any issues with the robustness test and thus proves to be able to handle frequent and especially infrequent verbs in passive sentences. The BiLSTM model scored a fail rate of 12%, misclassifying 60 out of 500 samples. The main issue appeared to be the verb *cuddled* - which can also be seen in Section 6.3. In fact, all sentences that were mislabeled contained the verb *cuddled*. The following list provides an overview of typical errors:

1. Error 1: [ARG0: Carl] was [V: cuddled] [ARG0: by Peter] .
2. Error 2: [ARGM-MNR: Kathy] was [V: cuddled] [ARG0: by Peter] .

The first error misclassifies the ARG1 with an ARG0, which could already be observed in Section 6.3. This might be because the model is more trained on active sentences and thus expects an ARG0 at the beginning of the sentence. The second common error was labeling ARG1 as ARGM-MNR. MNR specifies how an action is performed, which in this case could refer to the *cuddling* that Peter is doing. Overall, the performance of verb variation on passive sentences is fine, especially BERT does not have an issue with it and a fail rate of 12% on a sample of 500 is still a relatively small number. In the end, only one verb actually causes it. Removing it from the robustness test it could improve the performance, but in the end, a model should be able to work with verbs that are not as frequent as well.

6.4 Gardenpath Sentence

The gardenpath sentences are too complicated for both models and receive a 100% fail rate. This is most likely to its ambiguity as human readers are also not able to understand the sentence. It would be interesting to look into different gardenpath sentences if the issue is still prevalent or not.

1. BiLSTM
    (a) Error: [ARG0: Carl] [V: painted] [ARG1: the table] [ARG2: with cracks] .
2. BERT
    (a) Error: [ARG0: Amy] [V: painted] [ARG1: the ceiling] [ARG2: with cracks] .

The error for both is the same, as the models are not able to detect *the ceiling with*

cracks as ARG1 but separate it into two different ones. The issue arises not only in the ambiguity but also in the uncommon sentence structure that gardenpath sentences often appear in.

6.5 Transitive and Intransitive Verbs

Overall, the performance of the transitive and intransitive verbs scored a low fail rate. BERT did not run into any problems, whereas BiLSTM had issues with TI1, TI2, and I2. The errors can be seen below:

1. TI1
    (a) Error: [ARG1: Julia] [V: left] [ARGM-TMP: after cleaning up].
2. TI2
    (a) Error: [ARG0: He] [V: left] [ARG1: the phone] [ARGM-LOC: on the table].
3. I2
    (a) Error: [ARGM-DIS: Andrea] [V: sang] [ARG1: very nice] .

For TI1, *Julia* is somehow seen as an ARG1, it is the only error that happened within this capability check and can thus be seen as a random error with no actual bug behind it. TI2 had a difficult time classifying the sentence and wrongly labeled ARG2 often times as ARGM-LOC. This does make sense since it indicates a location where the item has been left at. For intransitive verbs, I2 with the included adjectives had one issue and misclassified ARG0 as ARGM-DIS.

6.6 Fairness

The fairness test on Western and Non-Western, in this case Syrian, names resulted in a variation of failure rates in both models. In the BiLSTM model, the SRL has a failing rate of 1%, 2 out of 200 samples, for the Western names and a rate of 5%, 10 out of 200 samples, for the Non-Western names. BERT has no errors for the Western names and a fail rate of 18%, 36 out of 200 samples, for the Non-Western names. Among others, the following sentences were mislabeled:

1. BiLSTM
    (a) Western Names: [ARG0: Julie] Richardson [V: believes] [ARG1: in Confucianism]
    (b) Western Names: [ARGM-DIS: Alfred] Lewis [V: believes] [ARG1: in Jain]
    (c) Syrian Names: [V:] [ARG1: Kahhalé believes in Islam] .
    (d) Syrian Names: [V: Isidore] [ARG1: believes in Buddhism] .
2. BERT
    (a) Syrian Names: [ARGM-DIS:] [ARG0:] [V: believes] [ARG1: in Hinduism] .
    (b) Syrian Names: [V:] [ARG1: Audo] believes in Christianity .
    (c) Syrian Names: [V:] believes in Atheism .

As it can be seen for the BiLSTM and Western Names the last name caused an issue for the ARG0 classification. In the first sentence, the model managed to mark Julia as ARG0 but not Richardson and in the second sentence, Alfred was labeled ARGM-DIS instead of ARG0 and Lewis was not labeled at all. It could be that the names both did not get detected because Lewis can also be read as a first name and Richardson has the same beginning as the first name Richard, thus the model does not recognize them as a connection with the first names. With the Syrian names, both systems run into more problems which can be due to the mixture of different alphabets the names are distributed in, as it can be seen in the list some names are written in the Arabic alphabet and some in the Latin. Additionally, both models have been trained on data containing the Latin alphabet. Both models have an issue with classifying the V correctly and oftentimes connect it to the first name. Since in Arabic, the word order is often Verb Subject Object, this misclassification is logical. Due to this failure, the rest of the sentences are also misclassified. However, it shows that the models are aware of Arabic sentence structure in general but are not able to combine it with English.
The religion did not seem to have a big impact on the performance of the ARG0 detection as both models provided errors in a variety of religions without a certain pattern to be detected. Finally, the fairness test shows that the models both have problems working with Arabic names written in the Arabic Alphabet in English sentences. This happens rarely, but in terms of inclusion and supporting diversity, it would be a nice add-on for models to be able to work with that.

6.7 Limitations

It has to be mentioned that the report has its limitations. Due to limited availability of time, only a handful of tests could be crafted even though Semantic Role Labeling offers so many more. Additionally, AllenNLP does not support Windows-based systems so a solution had to be found by using Ubuntu. Furthermore, just because a system has a 0% fail rate does not mean that it does not include bugs and issues. Many more tests can be done in order to go into more depth about the performance and ultimately improve it. Apart from that, CheckList might not be the best way to test a model's capability because it is not always transparent and difficult to implement own capability checks. It is also debatable if PropBanks annotations (Babko-Malaya, 2005) should be seen as the default as they do not cover every linguistic phenomena and also caused many predictions by the models to be wrong. CheckList sometimes limits crafted samples, which sometimes causes a test to only have a small number of 20 sentences. By increasing the number, more bugs could be detected. Another limitation is that only MFT tests have been used in this report, whereas insightful bug detection can also be done with INV and DIS. On an interesting note, it has been noticed that often times when parts of the sentences are wrongly labeled as ARGM-DIS by the BiLSTM, the name starts with an A. So to speak *Andrea, Alfred,* and *Alexandra*. It is an indication that the model might work with a shortcut, for instance comparing those names to *Alan*, which is mentioned as the discourse marker in the PropBank annotation guidelines.

7 Use Case

Semantic Role Labeling should be generalizable and adaptable to various domains. Many models are mostly just trained on news articles, books, or Wikipedia articles. To be able to work on a specific domain, the performance of an SRL has to be tested with a focus on the distinctive phenomena within the domain. Let us paint the picture, for a research project English doctors' notes are labeled with semantic roles. The data is mostly unlabeled and only a small amount is already labeled. The usual goal of SRL is to figure out *Who did what to whom*, whereas in the medical field it will be more like *what is done/happening to whom*. The data, due to its structure, is not going to include a lot of noise since doctors' notes are usually intended to be short and straightforward.

To estimate the performance of an SRL model the following capabilities should be tested: active and passive, gardenpath, predicate ellipsis, and argument ellipsis. Additionally, robustness tests on named entity recognition, incomplete sentences, and spelling mistakes should be conducted as well as a test that keeps bias and fairness in mind. The robustness tests are necessary because those notes often contain spelling errors or incomplete sentences. In addition to that, the doctors' notes are very domain-specific which is why a Named Entity Recognition on medical related words is extremely useful so the model is able to detect them as a whole entity and does not split them up into different sentiment roles. For the capabilities, a comparison between active and passive is useful as many sentences are phrased in a passive structure, for instance, *The medication was not tolerated by patient xy.*. The gardenpath sentences are useful as the notes often appear in an unusual structure that might not seem grammatically correct at first glance. In addition to that, the sentences might lack predicates and arguments and should be tested on it as well.

8 Conclusion

Finally, it can be said that based on the findings of this report, the BiLSTM is oftentimes performing worse than BERT on the Semantic Role Labeling task with a focus on the many capabilities tested here. It has several bugs that need to be fixed in order to improve its performance. In future projects, it would be very interesting to inspect the linguistic capabilities on a BERT SRL and a BiLSTM SRL in more detail as not all the bugs have been detected with this little experiment. It has to be mentioned again that, just because a model scored a low or non-existent fail rate it does not mean that it is free of bugs, errors, and biases. Much more can be uncovered by working with tools like CheckList extensively.

References

Babko-Malaya, O. (2005). Propbank annotation guidelines. *URL: http://verbs. colorado. edu*.

Baker, C. F., Fillmore, C. J., & Lowe, J. B. (1998). The berkeley framenet

project. In *Proceedings of the 36th annual meeting of the association for computational linguistics and 17th international conference on computational linguistics - volume 1* (p. 86–90). USA: Association for Computational Linguistics. Retrieved from https://doi.org/10.3115/980845.980860 doi: 10.3115/980845.980860

Fillmore, C. J., et al. (1976). Frame semantics and the nature of language. In *Annals of the new york academy of sciences: Conference on the origin and development of language and speech* (Vol. 280, pp. 20–32).

Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N. F., ... Zettlemoyer, L. S. (2017). Allennlp: A deep semantic natural language processing platform..

He, L., Lee, K., Lewis, M., & Zettlemoyer, L. (2017). Deep semantic role labeling: What works and what's next. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 473–483).

Kingsbury, P., & Palmer, M. (2002, May). From TreeBank to PropBank. In *Proceedings of the third international conference on language resources and evaluation (LREC'02)*. Las Palmas, Canary Islands - Spain: European Language Resources Association (ELRA). Retrieved from http://www.lrec-conf.org/proceedings/lrec2002/pdf/283.pdf

Ribeiro, M. T., Wu, T., Guestrin, C., & Singh, S. (2020). Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*.

Shi, P., & Lin, J. (2019). Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255*.

A Overview and Examples for Capabilities

Capabilities	**Example(s)**	**Name**	**Testing**
Conjunction	*(first name) met Max and smiled.*	C1	ARG0 predicate01
	John met (first name) and smiled.	C2	ARG1 predicate01
	(first name) met (first name) and smiled.	C3, C4	ARG0 and ARG1 predicate01
	(first name) met Max and smiled.	C5	ARG0 predicate02
Denotation	*John cooks pasta and Anna eats it.*	D1	ARG1 predicate02
Active and Passive	*(first name) cuddled/kissed someone.*	A1, A3	ARG0
	Peter cuddled (first name).	A2	ARG1
	(first name) was cuddled/hugged/adored by Peter.	P1, P2, P3	ARG1
	(first name) was (verb) by Peter.	P4	ARG1
Gardenpath Sentences	*John painted the wall with cracks.*	G1	ARG1
Transitive and Intransitive Verbs	*(first name) left (after cleaning up).*	TI1	ARG0
	He left (the object) on the table.	TI2	ARG1
	(first name) loves cats.	T1	ARG0
	She sang.	I1	ARG0
	She sang (adjective)	I2	ARG0
Robustness: Verb Variation	*(first name) was (verb) by Peter.*	P4	ARG1
Robustness:	*Peter does not/doesn't cuddle someone.*	A4, A5	ARG0
Negation Realization	*Peter cuddles no one.*	A6	ARG0
Fairness Test	*(western name) believes in (religion)*	F1	ARG0
	(non-western name) believes in (religion).	F2	ARG0

Table 2: Overview of tested capabilities with example sentence(s)