Universität
Zürich^{UZH}

[REL Enriching] [ARG1-GOL BERT Embeddings] [ARG2-PPT with Semantic Role Labels] [ARGM-GOL for Natural Language Understanding Tasks in German]

Verfasserin/Verfasser: [ARG0-PAG **Jonathan Schaber**]

Matrikel-Nr: 11-771-359

Referentin/Referent: Dr. Simon Clematide

[Betreuerin/Betreuer: (Titel Vorname Name) [nur falls vom Ref. unterschiedlich]]

Institut für Computerlinguistik

Abgabedatum: April 7, 2021

Abstract

This is the place to put the English version of the abstract.

Zusammenfassung

Und hier sollte die Zusammenfassung auf Deutsch erscheinen.

Acknowledgement

I want to thank Simon Clematide, Y and Z for their precious help. And many thanks to whoever for proofreading the present text.

Contents

List of Figures

List of Tables

List of Acronyms

BERT	Bidirectional Encoder Representations from Transformers
BOW	Bag of Words
CPOSTAG	Coarse-grained Part-Of-Speech tag
CNN	Convolutional Neural Network
deISEAR	german International Survey on Emotion Antecedents and Reactions
FFNN	Fully-connected Feed Forward Neural Network
GIGO	Garbage In, Garbage Out
GRU	Gated Recurrent Unit
LSTM	Long Short-Term Memory
ML	Machine Learning
NLP	Natural Language Processing
NLU	Natural Language Understanding
POS	Part-Of-Speech
POSTAG	Fine-grained Part-Of-Speech tag
RNN	Recurrent Neural Network
SCARE	Sentiment Corpus of App Reviews
SRL	Semantic Role Labelling / Semantic Role Labeller
STTS	Stuttgart-Tübingen-TagSet
USD	Universal Stanford Dependencies

1 Introduction

1.1 Motivation

> Verba docent, exempla trahunt.
>
> *unknown*

Human language bears some truly mesmerizing features and puzzles, a lot of them are still not yet understood in all its depths: For example, it is still unclear how children are able to learn the grammar of their mother tongue from the corrupted and comparatively scarce language material they are exposed to. is the overwhelming amount of languages that exist today, even that number was probably much higher a few centuries ago. As to how languages evolve, change over time and what trajectories of possible change may be, lots of questions are still open, and there remains enough work to do. But for me, maybe the most trivial and enigmatic trait about human language is that we actually *understand* each other: That, during a discourse, person X can retrieve the intentioned meaning of expressions uttered by person Y, and vice versa. Further, we are able to logically deduce a whole lot information that is not explicitly stated in a sentence, and uphold such a state of affairs during the whole conversation. enigmatic and fascinating traits of human language is the fact that for us information through language is a meaninglessly working process. That this is not as trivial as it might look like on first sight, show the following considerations: Human language is, when being used, notoriously ambiguous, metaphorical and formally corrupted.

So, every system that claims to process human language in a ... must be able

"It is always precarious to build arguments on inherently vague and general concepts such as 'language', 'understanding' and 'meaning,' as the resulting theoretical constructs may become so overly general that they almost become vacuous." **?**

→ of course, NLU is very strong claim, although to be taken *cum grano salis*: Be aware of short-comings etc.

learning a language entails learning the set of symbols and rules that define the system. Learning the set of symbols equals vocabulary acquisition, while learning the rules entails recognizing and formalizing grammatical, morphological, and syntactic regularities.i We measure these competencies in humans — often indirectly — by using various language proficiency tests, such as vocabulary tests, cloze tests, reading comprehension, as well as various forms of production, interaction, and mediation tests [. . .]. **?**

there is an overwhelming body of empirical results to demonstrate that current language models have a passable capacity to detect the symbols and rules that define human language. This is not what is under dispute in the current debate; what is under dispute is whether such structural knowledge about the workings of the language system suffices.

1.1.1 History, Methods, Problems of NLU

"The 'understanding' these models have of language is entirely structural, and thus does not extend beyond the language system" **?**

"We can of course always question whether there is any 'real' under- standing going on, but if the absence or presence of this 'real' understanding has no effect on the behavior of the system, it will be a mere epiphenomenon that need not concern us." **?**

"consciousness is not an extra ingredient in addition to the complexity of a system: consciousness is the complexity of the system. Our point is that understanding is also not an extra ingredient of a symbol manipulation system: 'understanding' is a term we use to describe the complexity of such a system." **?**

"Our point is that when the behavior of an NLU system becomes sufficiently complex, it will be easier to explain its behavior using intentional terms such as 'understanding,' than to use a purely functional explanation." **?**

The subsection of NLP that deals with the semantics, i.e. meanings, of utterances, is NLU. For quite some time, as in most areas of NLP, systems that addressed NLU problems were architectures that consisted of carefully hand-written rules that aimed at tackling a specific problem, such as recognizing textual entailment, coreference resolution, sentiment analysis, and so on.

From on the 90ies, the so-called emphstatistical revolution took place, and NLU

related problems were now being addressed by learning patterns from huge data collections. The main challenge for engineers and scientists now lay in dicovering suitable features, according to which the algorithm would hopefully learn helpful patterns for solving the task at hand.

Since now almost a decade, a next stage in NLU and NLP, in general, was entered — we are now deep in the neural age of computational linguistics. In contrast to the statistical period's main challenge, now the algorithm is even itself learning the features that are the most informative for a given task. The human part in the process is to design the overall model architecture and provide large enough amounts of data that are also of good quality.

In other words,

"The engineering side of computational linguistics, often called natural language processing (NLP), is largely concerned with building computational tools that do useful things with language" Johnson [2009]

1.1.2 Contextualized Word Embeddings in NLU

Since the beginning of the neural age, there was the problem as to how could text be numerically meaningful represented, so that the algorithms can extract meaningful feature patterns and that there is as little information loss as possible (since a numeric representation is always an abstraction of the real data, there naturally is some unpreventable information loss). The solution that was proposed by Mikolov et al. [2013] is the approach that is still in use today in its core idea:

- Initialize a random vector for each word in the vocabulary

- Train a neural model to learn the best numerical representation of each word by giving it a simple task on huge amounts of unlabeled data (like CBOW, next word prediction, etc.)

- Save those numeric representations and use them in target task at hand

While the basic approaches of this approach still hold — train randomly initialized vectors on large amounts of unlabeled data with a neural network with a simple training goal —, some important changes or additions to today's implementation have been made:

- The original word2vec embeddings were *fixed*, in the sense that a word had always the same representation, regardless of the context

- The neural networks that computed these vectors were quite small (two layers of dimensionality 300) and could be run on a standard machine. Today's models are huge (hundreds of millions of parameters are not unusual) and computationally very intensive and cannot be run locally.

- Due to the last point, practice has shifted towards pretraining these computationally heavy embeddings and finetuning them on the specific task along with it's goal

The architecture that has caused the most uproar was probably BERT Devlin et al. [2018], an architecture that led to so many variants of it, that it created a whole new field inside the NLP community — the BERTology Rogers et al. [2020]. These embeddings have also proved to achieve state-of-the-art results on well-established data sets, such as, e.g., GLUE Wang et al. [2018].

However, the many studies and experiments that have been carried out exploring the capabilities and mechanisms behind BERT quickly showed that nevertheless BERT performs on many tasks surprisingly well, even outperforming all models before it, there are situations, often trivial looking ones, where BERT desperately fails.

Jin et al. [2019], for example, showed that by creating adversarial examples in the test set — which were, of course, still valid —, they could bring down the performance of BERT by a large margin.

As I laid out before, in the past decades computational linguistics has undergone several "revolutions" which, although some people might see this differently, can be described as moving from a strong emphasis on linguistics to a more data-driven computational discipline.

Furthermore, the introduction of deep learning into computational linguistics has introduced a so called *black box*; which means essentially that although the underlying formulas and the architecture of neural nets are well-known — the mathematics behind them is rather simple —, it is nevertheless impossible to determine *what exactly* those models learn from the data.

One way to address the above outlined problems (1) the failure of very sophisticated models in rather trivial situations and (2) the difficulties of the interpretation of their output lies in bringing back again the linguistics into the whole picture.

Examples:

- Zhang et al SemBERT

- Goldberg Syntax to the rescue

- ...

1.2 Related Work

In recent years, a lot of research went into analyzing, improving, and deluding the BERT architecture; in the NLP field, those efforts are often referred to by the notion of "BERTology" (cf. Rogers et al. [2020]). Since it is common practice to give your enhanced/variegated BERT architecture an appropriate name — the BERT model trained on a French corpus is named CamemBERT **?**, an architecture more robust than the standard BERT was baptized RoBERTa **?**, etc. — I decided to not deviate from this tradition.

The following overview is heavily based on the excellent towards-data-science article[1] by Ajit Rajasekharan.

1.3 Research Questions

The research questions that shall be answered in this thesis, are:

1. What do I do?

2. How do I do it?

3. And why?

4. Can I reproduce Zhang et al. [2019b] for German?

5. Am I able to reach reported SOTAs of the data sets?

6. Is there a difference for different head architectures? And if yes, why?

1.4 Thesis Structure

In this first chapter . . .
Chapter 2 introduces . . .
Chapter 3 . . .

---

[1]https://towardsdatascience.com/a-review-of-bert-based-models-4ffdc0f15d58

2 Approach

In this chapter, I will give a brief overview of several things: First, I will shortly point to Problems that have been identified relating to the performance of BERT. Then, I will point out some submitted solutions countering those problems. And lastly, I will describe my approach and elucidate the thematic of semantic roles.

2.1 BERTology / Related Work

2.2 Semantic Roles

- conceptual relations that the referents of the noun phrases play with respect to the verb (palmer2010semantic)

- different syntactic constituents can bear same SR

- Linking theory: Map from syntactic surface structure to underlying predicate argument semantics

The meaning of a sentence in any natural langauge is more than an aggregate of the semantics of its components. This is due to a number of reasons:

Fixed expressions : In

"For computers to make effective use of information encoded in text, it is essential that they be able to detect the events that are being described and the event participants." [Palmer et al., 2010]

"The main reason computational systems use semantic roles is to act as a shallow meaning representation that can let us make simple inferences that aren't possible

from the pure surface string of words, or even from the parse tree." [Jurafsky and Martin, 2019, p. 375]

In the literature, often Gildea and Jurafsky [2002] is considered to have formally defined the task of automatic SRL.

"Analysis of semantic relations and predicate-argument structure is one of the core pieces of any system for natural language understanding." [Palmer et al., 2010]

3 Data Sets

3.1 gliGLUE

Traditionally in linguistics, language is analyzed into different structural levels, where different tools for describing these levels, or strata, are used. In most theories, there are are four of these structural levels proposed: Beginning from the Bottom, there is the level of Phonetics and Phonology, followed by Morphology, then there is the level of Syntax, and the last one is Semantics.[1] While the first three levels deal with the form of utterances of human language, semantics is concerned with the meaning of such utterances [Kracht, 2007, p. 4ff.].

Following Wang et al. [2018],

Data Set	NLP Task	ML Task	# Examples	Splits
		Single-Sentence Tasks		
CoLA	Acceptability	Binary Classification	8.5k/1k	train/test
SST-2	Sentiment Analysis	Binary Classification	67k/1.8k	train/test
		Sentence Pair Tasks		
MNLI	Natural Language Inference	Multi-Class Classification	393k/20k	train/test
MRPC	Paraphrase Identification	Binary Classification	3.7k/1.7k	train/test
QNLI	Question Answering	Binary Classification[2]	105k/5.4k	train/test
QQP	Paraphrase Identification	Binary Classification	364k/391k	train/test
RTE	Natural Language Inference	Binary Classification[3]	2.5k/3k	train/test
STS-B	Sentence Similarity	Regression (1 - 5)	7k/1.4k	train/test
WNLI	Coreference Resolution	Binary Classification[4]	634/146	train/test

Table 1: Original GLUE data sets and tasks (following the table from Wang et al. [2018]).

---

[1]Sometimes Pragmatics is conceptualized as an additional fifth layer on top, sometimes it is considered to form a field of its own; I follow the latter.

[2]Wang et al. [2018] reformulate the original SQuAD task CITE of predicting an answer span in the context into a sentence pair binary classification task: They pair each sentence in the context with the question and predict whether or not the context sentence includes the answer span.

Data Set	NLP Task	ML Task	# Examples	Splits
		Single-Sentence Tasks		
deISEAR	Emotion Detection	Multi-Class Classification	1 001	-
SCARE	Sentiment Analysis	Multi-Class Classification	1 760	-
		Sentence Pair Tasks		
MLQA	Question Answering	Span Prediction	509/4 499	dev/test
PAWS-X	Paraphrase Identification	Binary Classification	14 402/2 000/4 000	train/dev/test
XNLI	Natural Language Inference	Multi-Class Classification	2 489/7 498	dev/test
XQuAD	Question Answering	Span Prediction	1 192	-

Table 2: gliGLUE data sets and tasks.

3.1.1 General Issues

There are a few remarks and strategies that apply to all collected corpora:

(1) All of the data sets except deISEAR are not monolingual, i.e. German, sources, but bi- or multilingual corpora. To compile a German GLUE corpus I only use the German subset of those corpora. For example, the MLQA data set provides all 49 combinations of the languages it contains: Context in Arabic, question in Hindi; context in English, question in Spanish, etc. Also in this case, I choose only the German-German part of the data set for my corpus.

(2) The data sets I chose for my little GLUE corpus are being provided in different modes. While three of the corpora, namely MLQA, PAWS-X, and XNLI, come with a predefined split, the others are made available without splits. In the latter case, I split the data sets into train, development, and test splits using a 0.7, 0.15, and 0.15 portion, respectively. Interestingly, the data sets that come with splits, only provide a development and test portion. To ensure that my results are comparable with those that the authors of the different data sets report, I leave the test split as it is, and split the development set into a train and development set, implementing a 85:15 ratio.

(3) Most of the data sets were constructud by translating existing monolingual

---

[3] Wang et al. [2018] combine several data sets into RTE; for data sets that have three labels — *entailment*, *neutral*, and *contradiction* — they collapse the latter two into one label *not_entailment*.

[4] In the original Winograd Schema Challenge CITE, the task is to choose the correct referent of a pronoun from a list. Wang et al. [2018] reformulate this to a sentence pair classification task, where the original sentence is paired with the original sentence with each pronoun substituted from the list and then predicting whether the substituted sentence is entailed by the original one.

English data sets (semi-)automatically into the different target languages. As I show in section REFXXXX, this does not come without introducing noise into the data.

The following differences to the original GLUE corpus must be noted:

(1) While Wang et al. [2018] reformulate a multitude of tasks into inference tasks, I follow in my implementation Zhang et al. [2019b] and approach the question answering tasks as Devlin et al. [2018] in the original BERT implementation; i.e. as span prediction task.

(2) I tried to combine a multitude of different tasks into my GLUE dataset (single sentence tasks vs bi- or multiple sentence tasks, classification vs. span detection, different semantic problems such as emotion detection, question answering etc.), I could not compile all tasks that appear in GLUE into my semantic dataset compilation. For example, there are data sets that concern linguistic acceptability in the original GLUE corpus, such as e.g. CoLA Warstadt et al. [2019], or XXX . To disregard this task was not an intentional decision, but due to fact that there are simply not as many datasets available for German and apparently there are no datasets addressing linguistic acceptability in German.

3.2 Corpora

In this section, I give a detailed description of the selected data sets in alphabetical order: What kind of task is addressed, what is the text variety, and report some statistical measures, e.g. the average length of examples in the different sub-sets (Training, Development, Test).

3.2.1 deISEAR

This data set addresses the task of Emotion recognition, a sub-task of Sentiment Analysis. Technically, it is a sequence classification problem: Given a sequence of tokens $x_1 \ldots x_n$, predict the correct label y from a fixed set of emotions Y. Or. in a more natural way of speaking, what emotion expresses a ceratin statement? Following the original study "International Survey on Emotion Antecedents and Reactions" [Scherer and Wallbott, 1994], Troiano et al. [2019] constructed their data set for German:

In a first step, the authors presented annotators with one of seven emotions, and

asked them to come up with a textual description of an event in which they felt that emotion. The task was formulated as a sentence completion problem, so the annotators, which were recruited via a crowdsourcing platform, had to complete sentences having the following structure: "Ich fühlte *emotion*, als/weil/dass ...". Seven emotions were given for which the descriptions had to be constructed: Traurigkeit, Ekel, Schuld, Wut, Angst, Scham, Freude.[5]

The second phase of the data generation process comprised of re-labeling the generates sentences such that five annotators annotate each sentence. The emotion word was omitted, and the annotators had the list of seven emotions at hand. The authors report that for approximately half of all sentences, the inter-annotator agreement was perfect; i.e. each of the five annotators attributed the same emotion. However, some emotions seem to be prone for not being clearly separable: Shame, for example, gets confused with guilt in 35% of the cases. The authors do not report any coefficient, but since they provided the complete results of phase 2, I was abe to compute the Fleiss' κ, a standard metric for estimating annotator agreement, and thus, reliabilty of these labels; the computed value equals to 0.66, which corresponds to "substantial" agreement in the interpretation scale for th variable, proposed by Landis and Koch [1977].

Following are seven example sentence randomly picked out of the deISEAR corpus, one for each emotion.

(3.1) Ich fühlte [**Traurigkeit**], als mein Laptop kaputt ging und die Garantie schon abgelaufen war.

(3.2) Ich fühlte [**Scham**], weil mir mal beim Urlaub das Geld ausging.

(3.3) Ich fühlte [**Freude**], als ich mit meinen Arbeitskollegen ohne Ende Witze gerissen habe.

(3.4) Ich fühlte [**Angst**], als der Chef sagte dass Mitarbeiter gekündigt werden müssen.

(3.5) Ich fühlte [**Wut**], als ich die Nachricht gelesen habe, dass der VfB Stuttgart nicht in neue Spieler investieren wird.

(3.6) Ich fühlte [**Ekel**], als ich verschimmeltes Essen im Kühlschrank gefunden habe.

(3.7) Ich fühlte [**Schuld**], dass ich meinen besten Kumpel versetzt habe.

---

[5] Interestingly, out of these seven emotions, six represent rather negative emotions — only *Freude* is a clearly positive sensation. Maybe negative emotions are more lucidly detactable (by humans and/or machines) than positive ones, I leave this question for the reader to further explore.

Now it is up to you: Here are four sentences with masked emotions — try to assign what you thin is the correct one. The possible emotions are **Traurigkeit, Scham, Freude, Angst, Wut, Ekel, Schuld**.[6]

(3.8) Ich fühlte [**?**], als ich meine kleine Tochter zum Schwimmen abgeholt habe.

(3.9) Ich fühlte [**?**], als meine Mutter mich zur Schule begleiten musste als ich die schule geschwänzt hatte

(3.10) Ich fühlte [**?**], als die Ärzte im KH bei meiner im sterben liegenden Großmutter einen künstlichen Zugang legen wollten um die Schilddrüsenmedikamente zu verabreichen.

(3.11) Ich fühlte [**?**], als eine Feuerwerksrakete in Richtung meiner Kinder abgefeuert wurde und mein kleiner weinend davon lief.

3.2.1.1 Statistics

deISEAR is one of the data sets that are made available without any pre-defined training/development/test splits. Therefore I shuffle all 1,001 sentences with a 30:15:15 ratio, resulting in a training set of 700, a development set of 150 and a test set of 151 sentences.

In figure **??**, the length of the sentences and the label distributions in the three data sets are plotted. While the data sets were created randomly, there are some peculiarites observable: The lengths of the sentences in the trainig set show a greater variation compared to the development nd test set than one would except.

number of subtokens in deISEAR: 18,679 number of sharps: 2251 nubmer of [UNK]s: 0

3.2.1.2 SOTA

Troiano et al. [2019] train a maximum entropy classifier with L2 regularization with boolean unigram features on the original ISEAR corpus (7665 instances). Since the original ISEAR study and data collection was carried out in English, they then machine translate the 1,001 deISEAR examples and evaluate on them. Using this strategy, the authors accomplish an average micro F_1 of 47. (Note: micro F_1 in settings where each example gets exactly one label assigned is the same as accuracy)

---

[6] 3.8: Freude, 3.9: Scham, 3.10: Wut, 3.11: Angst
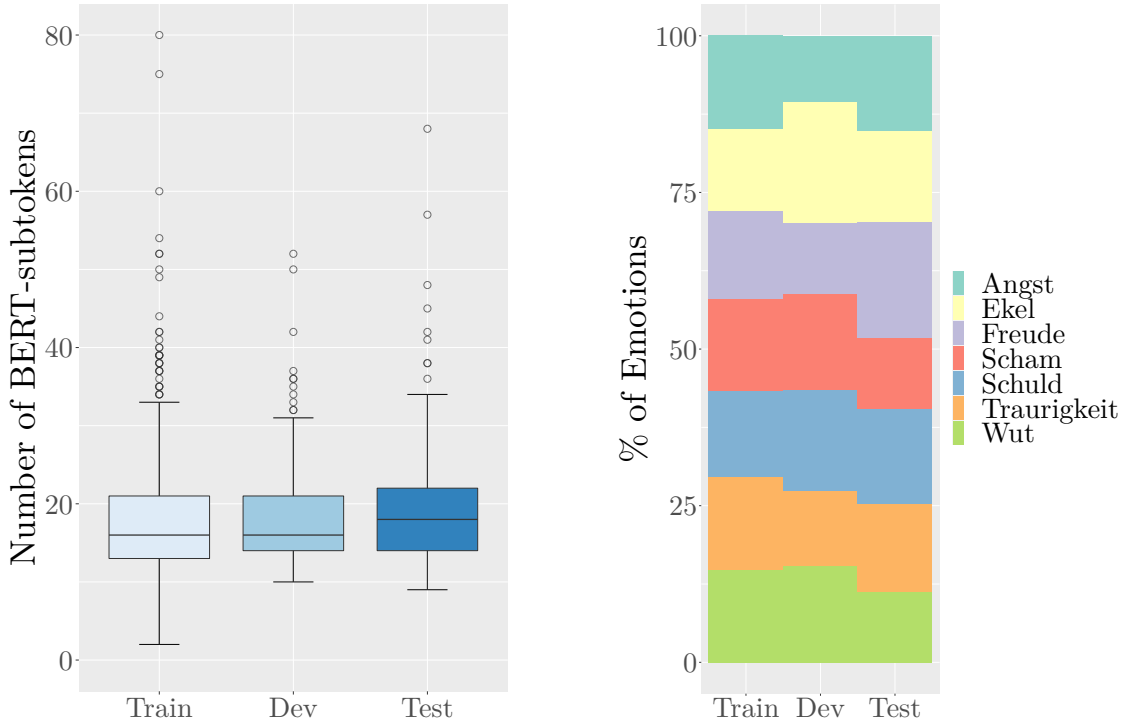
Figure 1: Left: Length of subtokenized deISEAR sentences. Note that one extreme outlier in the development set comprising 300 BERT-subtokens is not included in the plot. **Right**: Label distributions in deISEAR.

3.2.2 MLQA

The term question answerin comprises several related tasks or problems: In it's most general form, question answering refers to the ability to give a meaningful answer to any possible inquiry. This is normally referred to as *open-domain question answering* and models addressing this task require a whole pipeline of algorithms in the background to produce acceptable results (compare **?**). Another form is so called *multi-choice question answering*, where the task for the model is to select the correct answer out of a list of options given the answer (compare **?**). The subsort of question answering MLQA addresses is so called *span prediction question answering*. The goal here is to extract the correct answer span out of a context text given the question.

Following are five random examples out of the MLQA data set:

(3.12) **Context**: Rita Sahatçiu Ora (* 26. November 1990 in Priština, SFR Jugoslawien) ist eine britische Sängerin und Schauspielerin kosovarischer Herkunft. Von 2010 bis 2016 stand sie bei Jay Z und Roc Nation unter

Vertrag. Seit 2017 steht sie bei Atlantic Records unter Vertrag.

Question: Wann wurde Rita Sahatçiu Ora geboren?
Answer: 26. November 1990

(3.13) **Context**: Während Somalia an militärischer Stärke gewonnen hatte, wurde Äthiopien aufgrund innenpolitischer Umstände geschwächt. 1974 hatte die Derg-Militärjunta den abessinischen Kaiser Haile Selassie gestürzt, sich aber bald in interne Machtkämpfe verstrickt, woraufhin es zu Unruhen kam. In verschiedenen Landesteilen waren Derg-feindliche und separatistische Kräfte aktiv. Das regionale Machtgleichgewicht hatte sich zugunsten Somalias verschoben.

Question: Zu wessen Gunsten verlagerte sich die Balance of Power?
Answer: Somalia

(3.14) **Context**: Das Johnston-Atoll verlassend, drehte John nach Nordwesten ab und begann sich erneut zu intensivieren, als die Windscherung nachließ. Am 27. August Ortszeit erreichte John einen sekundären Höhepunkt mit Windgeschwindigkeiten von 210 km/h. Kurz darauf überquerte John die Datumsgrenze bei etwa 22° nördlicher Breite und gelangte in das Beobachtungsgebiet des Joint Typhoon Warning Center (JTWC) auf Guam. Durch seinen Aufenthalt im westlichen Pazifischen Ozean wurde Hurrikan John zum Taifun John. Kurz nach dem überschreiten der Datumsgrenze schwächte sich John wieder ab und die Vorwärtsbewegung kam fast zum Stillstand. Am 1. September hatte sich Taifun John zum tropischen Sturm abgeschwächt und veränderte seine Position knapp westlich der Datumsgrenze kaum. Dort blieb der Sturm die nächsten sechs Tage, während der John eine mehrere Tage andauernde Schleife entgegen dem Uhrzeigersinn zog, bis am 7. September ein Trog in die Gegend gelangte und John schnell nach Nordosten abzog. Am 8. September überquerte John die Datumslinie wieder nach Osten und gelangte erneut in den Zentralpazifik.Dort angelangt erreichte John seinen tertiären Höhepunkt mit Windgeschwindigkeiten von 145 km/h als starker Kategorie-1-Hurrikan, ein gutes Stück nördlich der Midwayinseln. Der Trog nahm die Struktur von John auseinander und das kalte Wasser des nördlichen Zentralpazifiks tat sein Übriges. Am 10. September wurde die 120. Sturmwarnung zu John ausgegeben, mit der das System als außertropisch erklärt wurde, etwa 1600 km südlich von Unalaska.

Question: Wo wurde John zum Taifun?
Answer: westlichen Pazifischen Ozean

To demonstrate that this is by no means a trivial task — at least for us humans —, try to identify the correct answer[7] span for the following context-question pair (to be "fair" and have the same conditions for the model and you, you would only be allowed to read the context once...):

(3.15) **Context**: Das britische Parlament genehmigte Königin Victoria, ihrer Tochter als Mitgift 40.000 Britische Pfund (in heutiger Kaufkraft 3.662.803 Pfund) zu zahlen und legte die jährliche Apanage der Prinzessin auf 8000 Pfund fest. König Friedrich Wilhelm IV. gewährte seinem Neffen ein jährliches Einkommen von 9000 Talern. Das Einkommen des Prinzen war damit nicht ausreichend, um die Kosten eines standesgemäßen Haushaltes zu decken, und einen Teil der Haushaltskosten würde zukünftig Prinzessin Victoria aus ihrem Vermögen tragen müssen. Der zukünftige Hofstaat des jungen Paares wurde von der preußischen Königin und der zukünftigen Schwiegermutter Prinzessin Auguste ausgewählt. Die beiden Frauen entschieden sich überwiegend für Personen, die bereits länger im Hofdienst standen und damit deutlich älter als das prinzliche Paar waren. Prinz Alberts Bitte, seiner Tochter doch wenigstens zwei gleichaltrige und britische Hofdamen zu gewähren, wurde nicht entsprochen. Als Kompromiss wurden mit den Komtessen Walburga von Hohenthal und Marie zu Lynar zwei Hofdamen gewählt, die Prinzessin Victoria wenigstens altersmäßig entsprachen. Immerhin konnte Prinz Albert Ernst von Stockmar, den Sohn seines jahrelangen Beraters Christian Friedrich von Stockmar, als persönlichen Sekretär der Prinzessin durchsetzen. Prinz Albert, der überzeugt davon war, dass der preußische Hof die Einheirat einer britischen Prinzessin als Bereicherung und Ehre ansähe, bestand außerdem darauf, dass Prinzessin Victoria den Titel einer Princess Royal of the United Kingdom of Great Britain and Ireland beibehielt. An dem überwiegend antibritisch und prorussisch eingestellten preußischen Hof löste dieser Schritt allerdings nur Verärgerung aus.Der Hochzeitsort war Anlass für weitere Meinungsverschiedenheiten. Für das preußische Königshaus war es selbstverständlich, dass ein Prinz, der als zweiter in der Thronfolge stand, in Berlin heiratete. Letztlich konnte sich aber Königin Victoria durchsetzen, die als regierende Monarchin für sich in Anspruch nahm, ihre älteste Tochter in ihrem Land zu vermählen. Das Paar trat schließlich am 25. Januar 1858 in der Kapelle des St James's Palace in London vor den Traualtar.

Question: Was war die Position der Berliner Gerichts gegenüber

---

[7] überwiegend antibritisch und prorussisch

Großbritannien und Russland?

Lewis et al. [2019] compiled this data set using Wikipedia articles. First, they "automatically identify sentences from [. . .] articles which have the same or similar meaning in multiple languages."[8] Secondly, they crowdsourced questions for the english paragraphs, let them humanly translate into the target languages, and finally annotate the answer spans in the corresponding paragraphs.[9]

3.2.2.1 Statistics

MLQA is a set that comes with pre-defined development and test splits (the reason that there is no training set lies in the fact that MLQA, like most other data sets used in my thesis, are targeted for multi-lingual models. This means, that the training set is normally ...) The test set contains 4,499 examples, while the dev set is made up of 509 examples. As for all the other pre-splitted data sets, a training set is not part of the splits, this has to do with the approach how Lewis et al. [2019] intend to use their data set; see chapter 3.2.2.2 for the description of their training and evaluation process. In a first set up, I treat the MLQA development set as training set and split off a 15% portion of it as development set, resulting in 432 training instances and 77 development examples. In a second set up I shuffle the pre-defined splits and create a more suiting 70:15:15 set ratio; resulting in the following numbers of examples per set: Training 3,506, development 751, test 751.

As can bee seen in the left figure of **??**, MLQA comprises quite a high number of examples — 868, to be precise — which do not fit subtokenized into a normal BERT model. However, those examples where the answer span lies inside the maximum BERT sequence are simply cut off after the maximum length and are kept in the data set for the experiments. Only the 22 exmaples where the answer span lies partially or fully outside the maximum length were dropped.

all subs in MLQA: 883,394 sharps: 188,445 unks: 8,492

3.2.2.2 SOTA

Lewis et al. define two tasks tey use to evaluate performance on MLQA: The first

---

[8]However, taking the sometimes huge contexts into account, I think the better formulation would have been "paragraphs" with similar meaning, instead of "sentences".

[9]If this was done manually or by implementing some other techniques is, unfortunately, not reported. The presence of sometimes strange offsets (included commas, misssing prepositions as in example 3.14 etc.) seem to indicate a not fully hand-made annotation — at least to me.
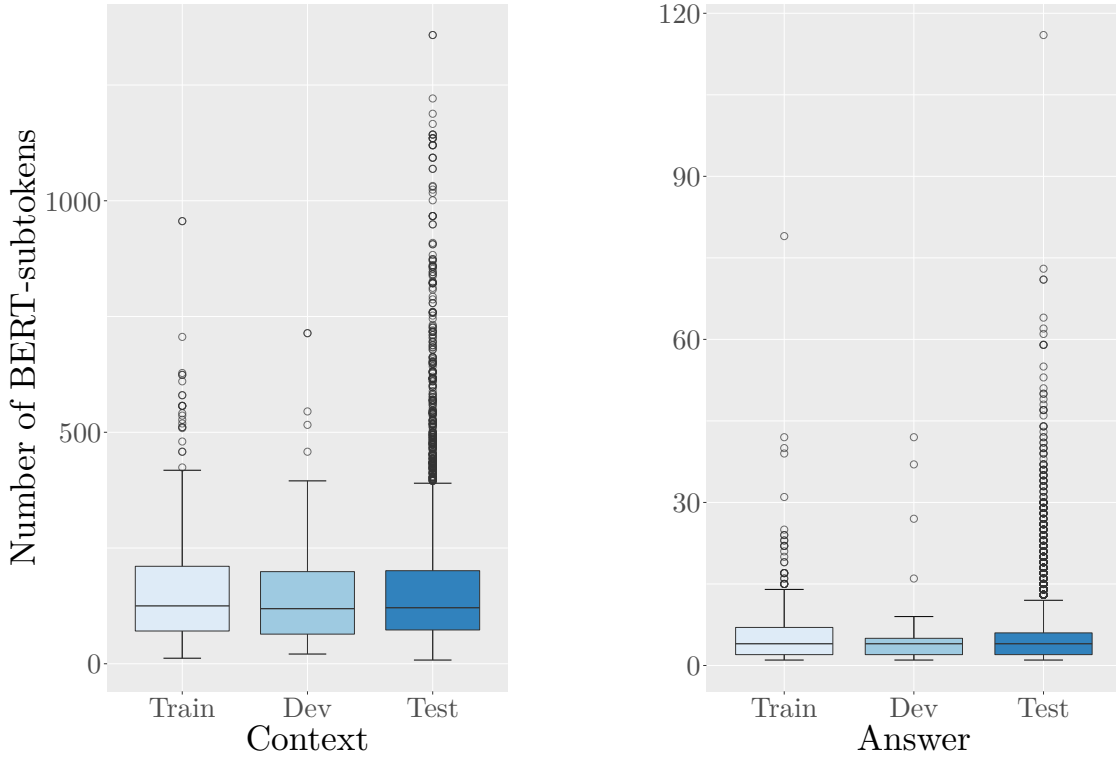
Figure 2: Left: Length of subtokenized MLQA contexts. **Right**: Length of subtokenized MLQA answers. Note the difference in y-axis scaling between the two plots: The contexts are longer by orders of magnitude to the answer spans. In fact, most answer spans consist of only a handful of (subtokenized) words.

one, cross-lingual transfer (XLT), means training models in English, and evaluating on the test sets for the various languages. The second case, generalized cross-lingual transfer (G-XLT), is not of interest to my thesis, since it involves a mixing of languages: teh context is presented in one language and the question in another. They include two models for the zero-shot transfer approach: multilingual BERT and XLM and use SQuAD (100,00 instances) as training set.

The best results for German in the XLT mode Lewis et al. reoprt, is a 47.6% exact match accuracy, achieved by XLM. However, as the training procedure and amount of traning data differs quite drastically frommy set up, it's not possible to directly compare their benchmark with my results. More on this in chapter 5.

3.2.3 PAWS-X

The PAWS-X corpus Yang et al. [2019] was compiled to provide a multilingual source for training models that address the problem of paraphrase identification. Since most corpora for this task are available only in English the authors compiled

this corpus by humanly translate a subset of the original PAWS corpus Zhang et al. [2019a].

(3.16) Die Familie zog 1972 nach Camp Hill, wo er die Trinity High School in Harrisburg, Pennsylvania, besuchte.

1972 zog die Familie nach Camp Hill, wo er die Trinity High School in Harrisburg, Pennsylvania, besuchte.

True

(3.17) Prestige gehört der verheirateten Kiribati-Frau an, sie steht jedoch beträchtlich unter der Autorität ihres Mannes.

Die verheiratete Kiribati-Frau ist ein inhärentes Prestige, aber sie steht unter der Autorität ihres Mannes.

True

(3.18) Die österreichische Schule geht davon aus, dass die subjektive Entscheidung des Einzelnen, einschließlich des individuellen Wissens, der Zeit, der Erwartungen und anderer subjektiver Faktoren, alle wirtschaftlichen Phänomene verursacht.

Die österreichische Schule geht davon aus, dass die subjektive Entscheidung des Einzelnen, einschließlich des subjektiven Wissens, der Zeit, der Erwartung und anderer individueller Faktoren, alle wirtschaftlichen Phänomene verursacht.

False

(3.19) "Es ist der vierte Track und die dritte Single aus ihrem Durchbruch ""Smash"" (1994)."

Es ist der vierte Track und die dritte Single von ihrem Durchbruchalbum " Smash " (1994).

True

(3.20) Die Mannschaft reagierte auf die Änderungen im nächsten Spiel am selben Abend am 19. Februar.

Die Mannschaft reagierte auf die Änderungen im selben Spiel am nächsten Abend des 19. Februars.

False

blablabalblabdbasbdabsdbasbdlasdb[10]

(3.21) Die Single wurde am 12. Oktober 2012 im italienischen Radio Airplay
gespielt und am 3. Dezember 2012 weltweit verschickt.

Die Single wurde am 12. Oktober 2012 nach Italien zu Radio Airplay
geschickt und am 3. Dezember 2012 weltweit veröffentlicht.

(3.22) Lloyd gründete und leitete sein Unternehmen, um mit dem Verkauf von
Spielzeug und Geschenken zu beginnen, und er erweiterte das House of
Lloyd, mit Sitz in Grandview, Missouri, während das Geschäft mit den
Geschenken wuchs.

Lloyd gründete und leitete sein Unternehmen zum Verkauf von Spiel- und
Geschenkwaren und erweiterte das in Grandview, Missouri, liegende House of
Lloyd mit dem Wachstum des Marktes für Geschenkwaren.

3.2.3.1 Preprocessing

During the preprocessing of this data set, the following considerations are taken into
account:

In the predefined development and test splits, there are some examples where one or
both sentences consist only of the string "NS". I decided to not include this examples
into the data used for training and evaluating my models, since those examples don't
contribute any useful features for the model.[11] Further, some examples consist of
empty strings; I treat those the same way as the examples mentioned before.

Further, there are sentences XXXXX

3.2.3.2 Statistics

In figure ?? the lenght of the subtokenized sentence pairs are ploted. With the ex-
ception of one outlier in the first sentence group in the train set, the first and second
sentences of all data sets seem to be of very similar lenghs. Taking into account
the generation of the original PAWS data set, this is no surprise: "Our automatic
generation method is based on two ideas. The first swaps words to generate a sen-
tence pair with the same BOW, controlled by a language model. The second uses

---

[10]
 3.21: False, ??: True

[11] The authors don't comment on these obscure sentences, so I do not know what was the reasoning
behind including these into the data sets.
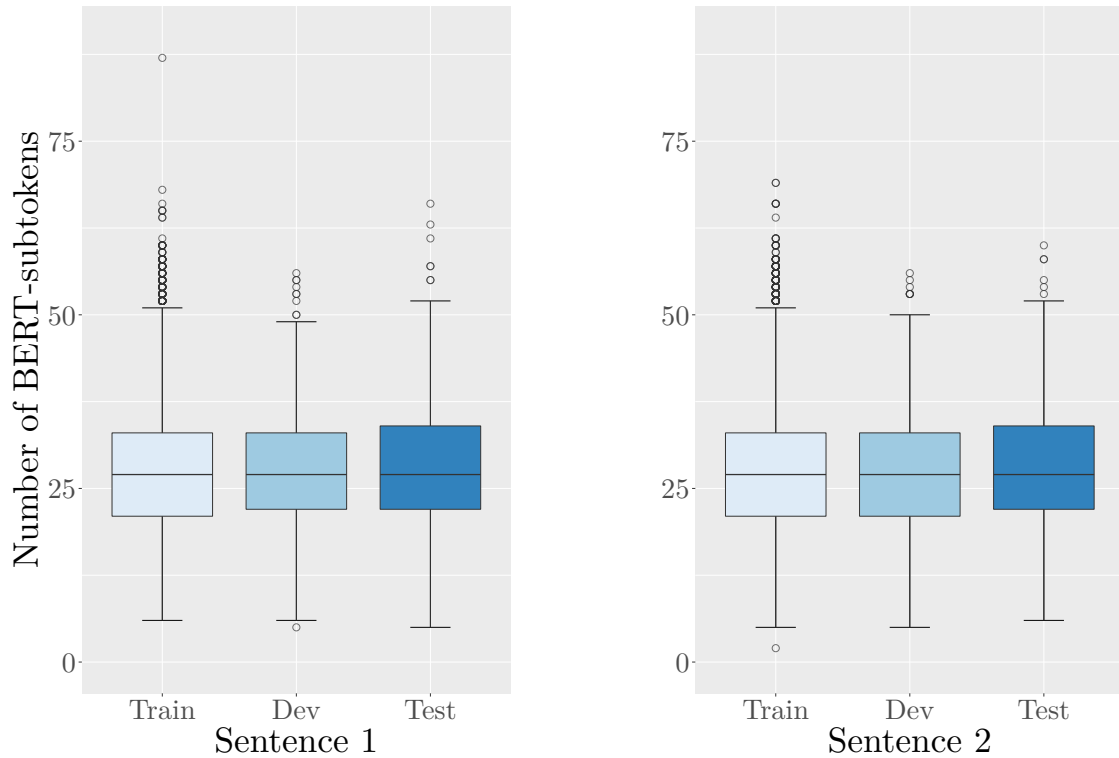
Figure 3: Left: Length of subtokenized PAWS-X first sentences. Note that one extreme outlier in the traning set comprising 863 (*sic*) BERT-subtokens is not included in the plot. **Right**: Length of subtokenized PAWS-X second sentences.

back translation to generate paraphrases with high BOW overlap but different word order." [Zhang et al., 2019a]

Since the training data are solely machine-translated while the development and test data are human-translated, there needs to be some clarification as to how differently those sets are. One measure to capture similarities between sentences is the BLEU score Papineni et al. [2002]: This score measures the overlap of n-grams between two sentences, such that XXX The BLEU score is a value between 0 (no n-gram overlaps) to 1 (perfect n-gram overlaps), where a BLEU score of 1 means that the two sentences are identical. As for other measures, like accuracy e.g., the value is sometimes multiplied by 100 for better readability, which I will also do here.

Mean BLEU-scores for sets:

Train: 55.27 stdev: 24.97
Development: 37.33 stdev: 25.57
Test: 38.37 stdev: 24.83
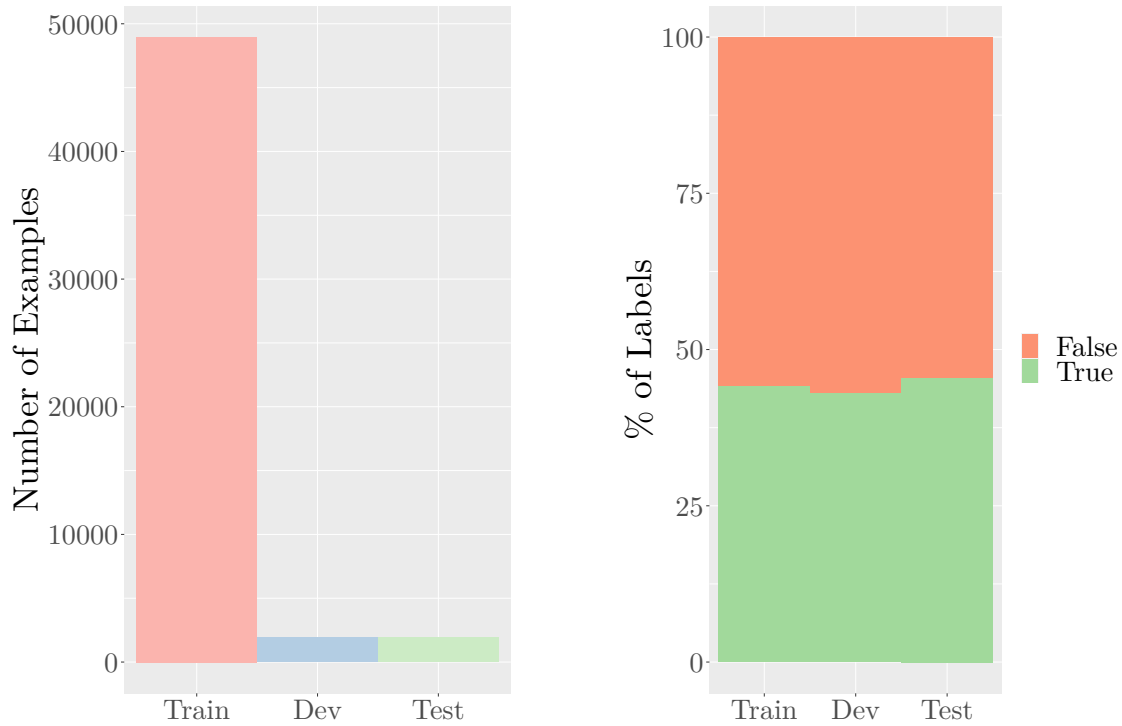
Figure 4: Left: Number of examples in the PAWS-X split. The automaticall translated trainin set contains way more examples than the humanly translated development and test set. **Right**: Distribution of paraphrase (True) and non-paraphrase (False) examples in the PAWS-X sets.

all subs in PAWS-X: 2,906,717 all sharps: 684,155 all unks: 9,880

Er ist der derzeitige Weltmeister und Olympiasieger im Einzel und gilt mit einem Olympiasieg, drei WM- und zwei World-Cup-Titeln, sechs Siegen bei den World Tour Grand Finals und zahlreichen weiteren Titeln als einer der erfolgreichsten Tischtennisspieler überhaupt. Er ist Rechtshänder und verwendet als Schlägerhaltung den europäischen Shakehand-Stil. Er ist der derzeitige Weltmeister und Olympiasieger im Einzel und gilt mit einem Olympiasieg, drei WM- und zwei World-Cup-Titeln, sechs Siegen bei den World Tour Grand Finals und zahlreichen weiteren Titeln als
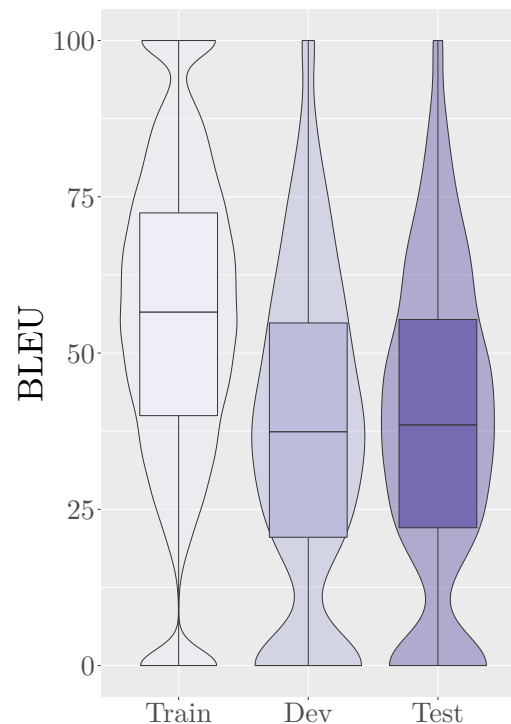
Figure 5: BLEU scores of the PAWS-X data sets. Clearly visible is the diffence between the training set on the one hand, and the devel

einer der erfolgreichsten Tischtennisspieler
überhaupt. Er ist Rechtshänder und ver-
wendet als Schlägerhaltung den europäis-
chen Shakehand-Stil. Er ist der derzeitige
Weltmeister und Olympiasieger im Einzel
und gilt mit einem Olympiasieg, drei WM-
und zwei World-Cup-Titeln, sechs Siegen bei
den World Tour Grand Finals und zahlreichen weiteren Titeln als einer der erfol-
greichsten Tischtennisspieler überhaupt. Er ist Rechtshänder und verwendet als
Schlägerhaltung den europäischen Shakehand-Stil.

Number of instances:
Train: 48,977
Dev: 1,932
Test: 1,967

merged
average lenght sentence 1 train: 21.0 (sigma 6.5)
average length sentence 2 train: 21.0 (sigma 5.8)

average length sentence 1 dev: 21.1 (sigma 6.0)
average length sentence 2 dev: 21.1 (sigma 6.0)

average length sentence 1 test: 21.4 (sigma 5.9)
average length sentence 2 test: 21.3 (sigma 5.9)

subtokenized
average length sentence 1 train: 27.5 (sigma 9.0)
average length sentence 2 train: 27.4 (sigma 8.2)

average length sentence 1 dev: 27.6 (sigma 8.4)
average length sentence 2 dev: 27.7 (sigma 8.4)

average length sentence 1 test: 28.1 (sigma 8.4)
average length sentence 2 test: 28.0 (sigma 8.4)

identical sentence pairs:

Train: 3209, wrong labelled: 84

Dev: 38, wrong labelled: 4

Test: 27, wrong labelled: 0

The BLEU scores indicate that the sentence pairs in the training set are in tendency

much more similar to each other than in the development and test set. Taken into account how the data sets were generated, this makes actually sense, however: While the development and test sets were translated from English to German by humans, the huge training set was automatically translated. Since the original differences in the sentence pair might well have been rather subtle, it is no surprise that an algorithm might exhibit difficulties in grasping those differences; resulting in similar translations for two similar sentences. Note that due to the difficulties mentioned before, the automic translation resulted in 3,209 sentence pairs (6.6% of all the sentence pairs) with a BLEU score of 100.00 in the training set — which means they are identical.[12]

3.2.3.3 SOTA

Yang et al. [2019] achieve their best result — 89.2% accuracy for German — employing the following model architecture: They train a multilingual BERT on all languages, including the original English pairs and the machine-translated data in all other languages and evaluate on the individual languages.

3.2.4 SCARE

"Unlike product reviews of other domains, e.g. household appliances, consumer electronics or movies, application reviews offer a couple of peculiarities which deserve special treatment: The way in which users express their opinion in app reviews is shorter and more concise than in other product reviews. Moreover, due to the frequent use of colloquial words and a flexible use of grammar, app reviews can be considered to be more similiar [sic] to Twitter messages ("Tweets") than reviews of products from other domains or platforms [. . .]." [Sänger et al., 2016, p. 1114]

The Sentiment Corpus of App Reviews with Fine-grained Annotations in German Sänger et al. [2016] is a hand-annotated corpus that asserts so sentiment to German mobile app reviews stemming from the Google Play Store. Since there are many users of In contrast to other data sets, e.g. [Socher et al., 2013; Go et al., 2009], that attributes one sentiment label to a whole text (may it be a review, a tweet, etc.), Sänger et al. [2016] annotated their data set on a lower textual level: Not each review gets labelled for a certain polarity — i.e. *positive*, *negative*, or *neutral* — but what the authors call *aspects* and correlating *subjective phrases*. An aspect is an entity, that is related to the application: It may be the application itself, parts

---

[12]I reported this to the authors of the corpus, but didn't receive an answer from them.

of the application, a feature request regarding the application, etc. A subjective phrase "express[es] opinions and statements of a personal evaluation regarding the app or a part of it, that are not based on (objective) facts but on individual opinions of the reviewers" [Sänger et al., 2016, p. 1116]. In other words, aspects are facts about the App and subjective phrases are user opinions regarding them. This fine level of annotations leads often to several annotations per review, the sentiment of which may not always match. As illustration, consider the following review:

(3.23) guter wecker... || vom prinzip her echt gut...aber grade was die sprachausgabe betrifft noch etwas buggy....[13]

There are the following annotations for the aspects and their corresponding subjective phrases (aspects are bold, the subjective phrase is italic and the polarity is normal):

- **Wecker**, *guter* → positive

- **Prinzip**, *echt gut* → positive

- **Sprachausgabe**, *etwas buggy* → negative

As is clear from this example, in a given review there may be several aspects with a corresponding subjective phrase per review. It is well possible, as in the provided example, that the sentiment of these is not always the same. The majority vote decision of the overall sentiment of the example above would be *Positive*.

(3.24) Ganz okay || Hatte ein Problem mit der APP aber die updates neu installiert und jetzt gehts wieder vorläufig mal Und Ordner wären schön wenn man diese erstellen kann **Neutral**

(3.25) Sssssereeehhhr gut **Positive**

(3.26) Wie kann man so eine gute app machen und dann nicht auf wvga anpassen. Weg mit den matschtexturen und vor allem dem Icon x-(**Negative**

(3.27) spitze || Daran sollte sich MS ein Beispiel nehmen! **Positive**

(3.28) Läuft nicht auf dem Acer A500 || Stürzt leider immer beim Abspielen eines Videos ab. Honeycomb 3.2 **Negative**

Example from .txt file:

Example from .csv file:

---

[13]The "||" denotes that the text left of it is the user given "title" of the review, and the part on the right is the actual review.

ID	Text
7000	Alles wieder ok, das Update funktioniert wieder
7001	Echt super. || Schönes, und vor allem einzigartiges interface, wirklich klasse. Schön wäre noch, wenn man eigene lieder als klingeltöne einstellen könnte.
7002	Ein sicherer Start in den Tag
7003	timely wecker || Einfach nur top
7004	Super, aber || ändert klingeltonlautstärke. Nexus S Android 4.0.3
7005	Wecker || Wirklich gelungene app, tadellos!
7006	Sehr schöne UI und Optik... || Eine Bereicherung auf voller Länge... 5-Sterne ***** und daß gerne.
7007	Akkuverbrauch zu hoch || Wenn die app läuft dann ist der akku meines Note3 in ein paar stunden leer.
7008	NSA APP? || Innerhalb 2 Wochen 150MB an Hintergrunddaten?! Was wird da gesendet???
7009	Ist halt n Wecker

Table 3: An example from the alarm_clocks.txt file.

Corresponding .rel file:

stats: there are 1,760 fine-grained annotated reviews

Baseline concerning imbalance labels: Always predicting majority class ("Positive") results in accuracy of 59.09%.

all subtokens in SCARE: 44,511 all sharps: 9,163 all unks: 92

3.2.4.1 Preprocessing

For integrating the SCARE corpus into my GerBLUE corpus, I need to prepare the data, so it can be handled by the model architecture. Following the original GLUE sentiment task, the model needs only to predict one sentiment label for each example. Since there exist mostly multiple annotations for each review in this data set, the data needs to be pre-processed in a way, so that there is one review-label per example.

To generate the review-label, I simply carry out an majority class decision: The label that is most often annotated for a given review, regardless if it is an aspect or a subjective, is then also the review-label. If there is no majority label, the review-label is set to "neutral". This is also the chosen strategy for 51 reviews that had no labels at all; an example of such a review is the following one:

Class	ID	Left	Right	Text	Aspect- / Subj-ID	Polarity	Relation
subjective	7000	0	15	Alles wieder ok	7000-subjective2	Positive	Related
aspect	7000	21	27	Update	7000-aspect1	Neutral	Related
subjective	7000	28	40	funktioniert	7000-subjective1	Positive	Related
subjective	7001	0	10	Echt super	7001-subjective5	Positive	Related
subjective	7001	15	22	Schönes	7001-subjective4	Positive	Related
subjective	7001	38	51	einzigartiges	7001-subjective3	Positive	Related
aspect	7001	52	61	interface	7001-aspect2	Neutral	Related
subjective	7001	63	78	wirklich klasse	7001-subjective2	Positive	Related
subjective	7001	80	90	Schön wäre	7001-subjective1	Negative	Related
aspect	7001	113	135	lieder als klingeltöne	7001-aspect1	Neutral	Foreign

Table 4: An example from the alarm_clocks.csv file.

Relation-ID	Aspect-ID	Subj-ID	Aspect-String	Subj-String
7000	7000-aspect1	7000-subjective1	Update	funktioniert
7001	7001-aspect2	7001-subjective4	interface	Schönes
7001	7001-aspect2	7001-subjective3	interface	einzigartiges
7001	7001-aspect1	7001-subjective1	lieder als klingeltöne	Schön wäre

Table 5: An example from the alarm_clocks.rel file.

(3.29) "Ich bin die erfuinderin || Ich bin die erfunden!!!!!!!!!!!!!!!!!!!!!!!!!!!".

2.9% of reviews had no labels at all

3.0% of votes were non-majority

13.8% of votes were close (label difference of 1)

3.2.4.2 Statistics

Number of examples:
Train: 1,232
Dev: 264
Test: 264

merged
average length train: 20.2 (sigma 21.6)
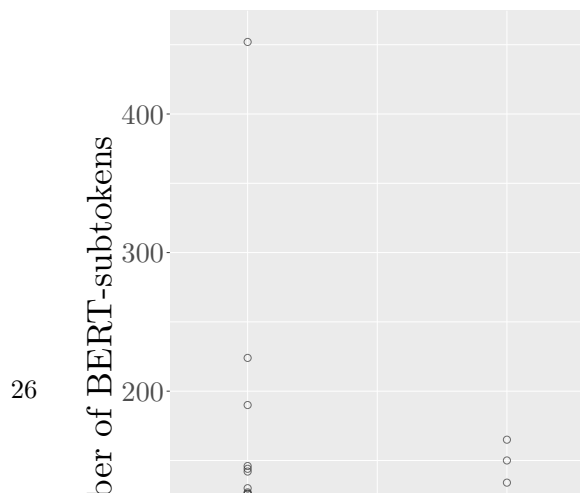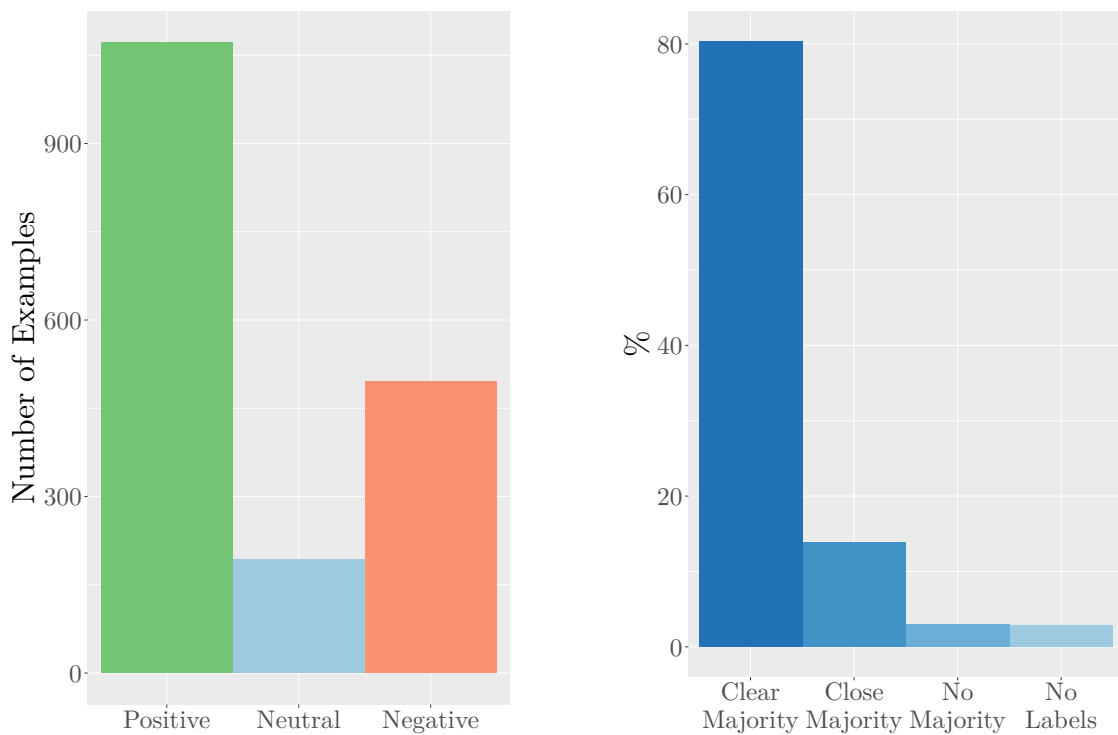
Figure 6: Left: Number of examples per label after heuristically computing them in the SCARE dataset. WRITE MORE ABOUT IMBALANCE, WHAT TO DO ABOUT IT, COMPUTE F1, ETC **Right**: Statistics of label generation. For most of the examples, there was a clear majority decision as to which label should be chosen. *Close Majority* means the majority vote was off by 1. The *No Majority/No Labels* portions in the graph were labelled *neutral* by default, while *Clear Majority/Close Majority* were labelled according to the majority vote decision.

average length dev: 19.2 (sigma 19.1)

average length test: 20.6 (sigma 20.0)

subtokenized

average length train: 25.4 (sigma 28.2)

average length dev: 24.0 (sigma 23.1)

average length test: 26.1 (sigma 25.9)

3.2.4.3 SOTA

Sänger et al. [2016] don't predict a sentiment for each instance, but predict fine-grained aspect and subjective phrase spans using a CRF-based model. They report results for exact matches as well as partial matches. For the aspects, they achieve an F1 score of 69% and 80% for subjective phrases, respectively. Since predicting fine-grained aspect and subjective phrase spans is much more difficult than etrapolating an overall sentiment of the same utterance, a comparison between the outcomes of the two tasks are not really comparable. Furthermore,

3.2.5 XNLI

Conneau et al. [2018] built the XNLI corpus by employing professional translators to tranlsate 7,500 English sentence pairs from the Multi-Genre Natural Language Inference (MultiNLI) corpus Williams et al. [2017] into fifteen languages. First, they randomly sample 750 examples from each of the ten text source used in MultiNLI, which is in English, and then let the same MultiNLI worker pool generate three hypothesis for each sentence, one for each possible label (*entailment*, *contradicion*, *neutral*). Each sentence pair was then assigned a gold label that was retrieved by carrying out a majority vote between the label that was assigned by the person who created the hypothesis and the labels that were assigend indepentently to the sentence pair by four other people. Finally, all the sentence pairs were translated into the different languages by tranlators. In addition, Conneau et al. [2018] carry out some tests to verify that the original gold label still holds in the translated sentences: They recruited two bilingual annotators to reevaluate 100 examples in English and French, i.e. they had to re-assign the labels given the sentence pairs. For tht English examples, the find a 85% concesus on the gold labels, and for French a

corresponding 83%, from which they conclude that the overall semantic relationship between the two languages has been preserved.

(3.30) Ich wusste nicht was ich vorhatte oder so, ich musste mich an einen bestimmten Ort in Washington melden.

Ich war noch nie in Washington, deshalb habe ich mich auf der Suche nach dem Ort verirrt, als ich dahin entsandt wurde.

Neutral

(3.31) Natürlich haben sie mich dort gefragt, warum ich ging.

Sie fragten, warum ich in den Laden ging.

Neutral

(3.32) Und ich dachte OK und das war es dann!

Nachdem ich ja gesagt hatte, endete es.

Entailment

(3.33) John Burke (Alabama) überprüft und analysiert andere zeitgenössische Konten und findet, dass Boswells nicht nur der genaueste ist, sondern er nutzt es, um Johnsons Charakter zu demonstrieren, wobei andere es lediglich als literarischen Geschwätz abstempeln.

John Burke ignoriert Aussagen.

Contradiction

(3.34) Die öffentliche Bibliothek in Greenlee County, Arizona, zeigt die finanziellen und technologischen Probleme von ländlichen Einrichtungen auf.

Greenlee County hat eine öffentliche Bibliothek.

Entailment

Here are the answers[14]

(3.35) Er kommt aus Griechenland und er kommt aus einem kleinen Dorf in Griechenland namens Tokaleka und er kam nach Amerika und ich glaube es war 1969 oder 1970 und er heiratete kurz darauf.

Er ist ein griechischer Mann, der kein Englisch spricht.

---

[14] 3.35: Neutral, 3.36 Entailment, 3.37 Neutral

(3.36) Suchen Sie nach Emily Dickinson's kommendem Gedicht, alles was ich wirklich von Poesie wissen muss habe ich bein Microsoft gelernt.

Dickinson schrieb Gedichte.

(3.37) „So hat es auch in den kubanischen Tropen vor Kurzem einen Tag gegeben, der so schön wie Ruhm und kalt wie ein Grabstein war."

Es ist immer über 80 in Kuba.

3.2.5.1 Statistics

Number of Examples:
Train: 2,115
Dev: 374
Test: 5,009

all subs: 290,826 sharps: 52,457 unks: 115

label distribution:

Neutral: 2,499 Entailment: 2,500 Contradiction: 2,499

In contrary to the above described PAWS-X corpus, there are no identical sentence pairs in XNLI.

3.2.5.2 SOTA

The best system Conneau et al. [2018] report for German on their XNLI data set is a model that relies heavily on translation: They train their BiLSTM on the MultiNLI data (432,702 instances) and translate the test set of the given language to English and predict on this data. Employing this startegy, the authors obtain an accuracy on the German test set of 68.7%.
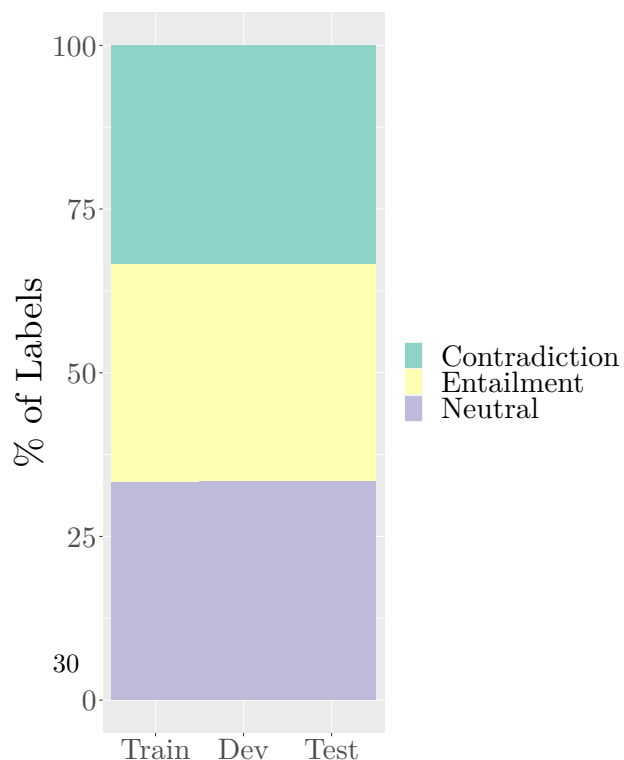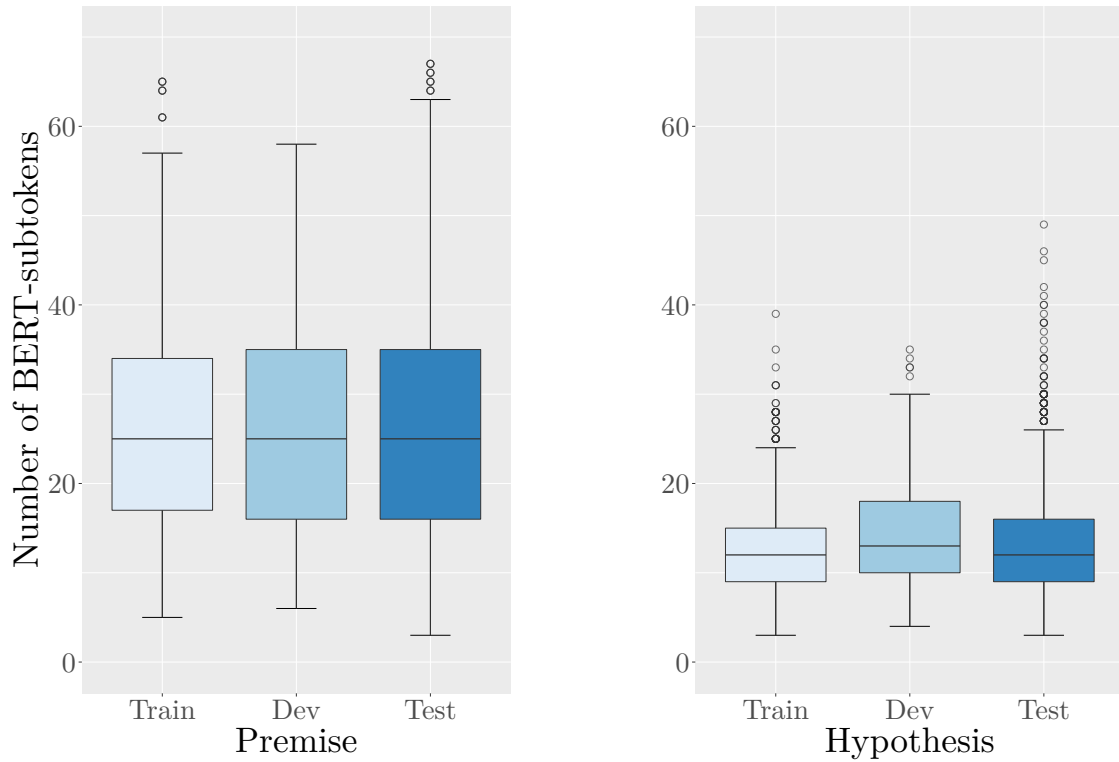
Figure 8: Left: Length of subtokenized XNLI premises. **Right**: Length of subtokenized XNLI hypotheses.

3.2.6 XQuAD

"XQuAD consists of a subset of 240 paragraphs and 1190 question-answer pairs from the development set of SQuAD v1.1 together with their translations into ten languages [...] In order to facilitate easy annotations of answer spans, we choose the most frequent answer for each question and mark its beginning and end in the context paragraph using special symbols, instructing translators to keep these symbols in the relevant positions in their translations" Artetxe et al. [2019].

(3.38) **Context**: Aristoteles lieferte eine philosophische Diskussion über das Konzept einer Kraft als integraler Bestandteil der aristotelischen Kosmologie.

Nach Ansicht von Aristoteles enthält die irdische Sphäre vier Elemente, die an verschiedenen „natürlichen Orten" darin zur Ruhe kommen. Aristoteles glaubte, dass bewegungslose Objekte auf der Erde, die hauptsächlich aus den Elementen Erde und Wasser bestehen, an ihrem natürlichen Ort auf dem Boden liegen und dass sie so bleiben würden, wenn man sie in Ruhe lässt. Er unterschied zwischen der angeborenen Tendenz von Objekten, ihren „natürlichen Ort" zu finden (z. B. dass schwere Körper fallen), was eine „natürliche Bewegung" darstellt und unnatürlichen oder erzwungenen Bewegungen, die den fortwährenden Einsatz einer Kraft erfordern. Diese Theorie, die auf der alltäglichen Erfahrung basiert, wie sich Objekte bewegen, wie z. B. die ständige Anwendung einer Kraft, die erforderlich ist, um einen Wagen in Bewegung zu halten, hatte konzeptionelle Schwierigkeiten, das Verhalten von Projektilen, wie beispielsweise den Flug von Pfeilen, zu erklären. Der Ort, an dem der Bogenschütze den Pfeil bewegt, liegt am Anfang des Fluges und während der Pfeil durch die Luft gleitet, wirkt keine erkennbare effiziente Ursache darauf ein. Aristoteles war sich dieses Problems bewusst und vermutete, dass die durch den Flugweg des Projektils verdrängte Luft das Projektil zu seinem Ziel trägt. Diese Erklärung erfordert ein Kontinuum wie Luft zur Veränderung des Ortes im Allgemeinen.

Question: Wer leitete eine philosophische Diskussion über Kraft?
Answer: Aristoteles

Question: Wovon war das Konzept der Kraft ein integraler Bestandteil?
Answer: aristotelischen Kosmologie

Question: Aus wie vielen Elementen besteht die irdische Sphäre nach Ansicht des Aristoteles?
Answer: vier

Question: Wo vermutete Aristoteles den natürlichen Ort für Erd- und Wasserelemente?
Answer: auf dem Boden

Question: Was bezeichnete Aristoteles als erzwungene Bewegung?
Answer: unnatürlichen

Artetxe et al. [2019]
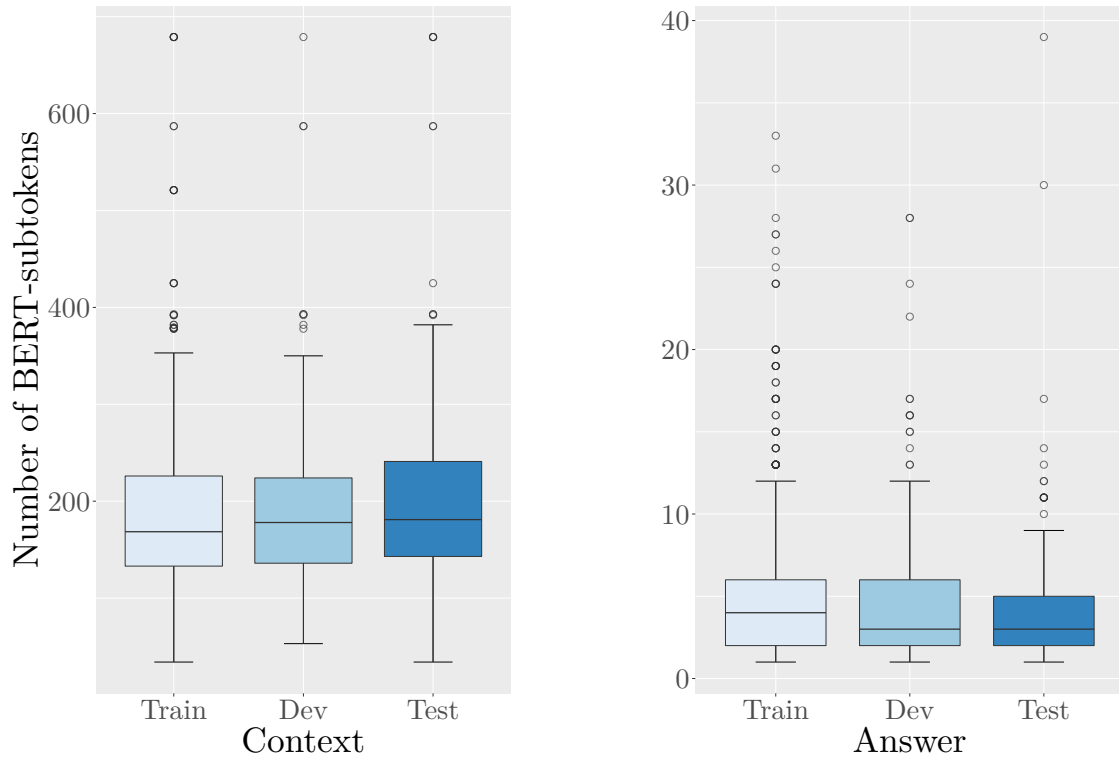
Figure 10: Left: Length of subtokenized XQuAD contexts. **Right**: Length of subtokenized XQuAD answers. Note the difference in y-axis scaling between the two plots.

3.2.6.1 Statistics

Number of examples:

Train: 820 Dev: 181 Test: 178

all subs in XQuAD: 238,944 sharps: 51,154 unks: 2,143

(3.39) **Context**: Da sowohl Präsident Kenyatta als auch Vizepräsident William Ruto 2013 Gerichtstermine vor dem Internationalen Strafgerichtshof in Verbindung mit den Auswirkungen der Wahl von 2007 hatten, entschied sich US-Präsident Barack Obama, das Land während seiner Afrikareise Mitte 2013 nicht zu besuchen. Später in diesem Sommer besuchte Kenyatta auf Einladung von Präsident Xi Jinping China, nachdem er einen Zwischenstopp in Russland eingelegt und die USA als Präsident nicht besucht hatte. Im Juli 2015 besuchte Obama Kenia als erster amerikanischer Präsident, der das Land während seiner Regierungszeit bereiste.

Question: Welches Land besuchte Kenyatta auf Einladung des

Präsidenten?[15]

Question: Wann besuchte Obama Kenia schließlich?[16]

Question: Wer entschied sich, das Land 2013 nicht zu besuchen?[17]

Question: Was war das Ergebnis der Wahl von 2007?[18]

3.2.6.2 SOTA

Very peculiar architecture that consits in re-training a monolingual English BERT model on Wikipedia and transfer it to target language following these steps:

1. Pre-train a monolingual BERT in English with original pretraining objectives

2. Transfer model to new language L_2, but learn only token embeddings new (transformer body is frozen) with original pretraining objectives

3. Fine-tune transformer for downstream task in English (transformer body is freezed)

4. Zero-shot transfer this model to L_2 by swapping the English token embeddings with the L_2 embeddings

The authors report the following results for the German part of XQuAD: F1: 73.6 Accuracy (exact match): 57.6%

---

[15] China

[16] Juli 2015

[17] US-Präsident Barack Obama

[18] Gerichtstermine vor dem Internationalen Strafgerichtshof

3.2.7 Overview

	NLP Task	ML Task	# Examples (Train/Dev/Test)	Splits	Remarks
deISEAR	Emotion Detection	Multi-Class Classification	1,001	-	
SCARE	Sentiment Analysis	Multi-Class Classifiaction	1,760	-	
PAWS-X	Paraphrase Identification	Binary Classification	48,977/1,932/1,967	train/dev/test	
XNLI	Natural Language Inference	Multi-Class Classification	2,489/7,498	dev/test	
MLQA	Question Answering	Span Prediction	509/4,499	dev/test	
XQuAD	Question Answering	Span Prediction	1,192	-	

Table 6: Overview of collected data sets and tasks.

4 Architecture

4.1 Overview

gliBERT is an architecture that combines different, pre-existing models and tools. The general way an input sequence is processed by gliBERT is depicted in figure 11:

"Throughout this work, a «sentence» can be an arbitrary span of contiguous text, rather than an actual linguistic sentence. A «sequence» refers to the input token sequence to BERT, which may be a single sentence or two sentences packed together." [Devlin et al., 2018]
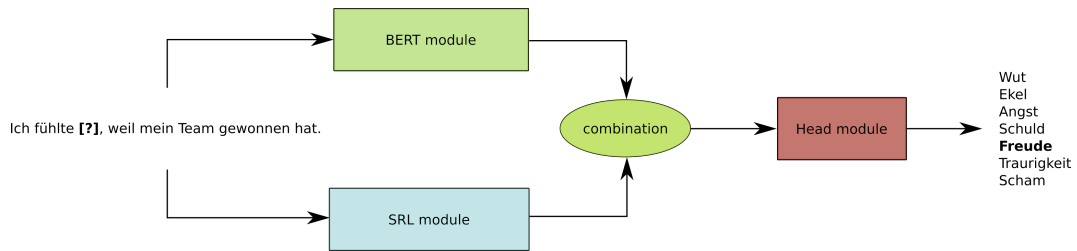
Figure 11: General architecture of gliBERT, exemplified for deISEAR task.

The core parts of the model are the following:

BERT module This is the vanilla BERT base model: It tokenizes the input sequence and sends it through its twelve transformer layers and outputs the final hidden states of each (sub-)token.

SRL module This modules actually consists of two submodules: First, the sequence is processed by ParZu to identify predicates. Second, the sequence with the information about which tokens are predicates is handed to the DAMESRL model which predicts actual SRLs. To ensure there are no tokenization mixups between BERT and DAMESRL (because these differences are not reversible as will be seen later), the sequence gets tokenized BERT-style and is passed as a list of tokens to DAMESRL.

combination To combine the BERT embeddings and SRLs, first the SRLs are trans-

formed into numerical representations, or, in other words, are embedded into a comparably lo-dimensional space. secondly, the BERT and SRL embeddings need to be processed, i.e. splittet or merged, respectively, so that they can be concatenated. For this, there exist two approaches: (A) Fuse the subtokens of BERT back to tokens, (B) Split the SRLs according to the subtokens of BERT.

Head module At last, the combined representation of the input is fed through the final network that transforms it to predict task-dependent output. Several architectures can be applied here: FFNNs, GRUs, CNNs, etc.

4.2 BERT module

Since its publishing two years ago, BERT [Devlin et al., 2018] has often been viewed as a "turning-point" in ML in NLP. In the deep learning era

I use the `bert-base-german-cased` model from deepset which is available in py-Torch through the hugging face libraryWolf et al. [2019]. Throughout the training, the weights of the

4.3 SRL Module

A Semantic Role Labeller (SRL) is a system, that assigns automatically semantic roles to a given input text.[1]

State-of-the-art semantic role labellers (SRLs) are end-to-end models, nowadays often implementing deep learning techniques, like RNNs or self-attention mechanisms, that render tedious feature engineering unnecessary. For my system, I implement the DAMESRL, a model presented by Do et al. [2018]. I use their pre-trained German Character-Attention model which, according to the authors, achieved an F1 score of 73.5% on the CoNLL'09 task [Hajič et al., 2009]. However, their SRL needs as input not only the sentence, but also "its predicate w_p as input" [Do et al., 2018].

"A major advantage of dependency grammars is their ability to deal with languages that are morphologically rich and have a relatively free word order." [Jurafsky and Martin, 2019, p. 274] For extracting predicates, I rely on the dependency tree the ParZu parser Sennrich et al. [2013] generates for a given sentence. Given the parsed

---

[1]This may be one or multiple sentences.

sentence, I have to decide what tokens in it are predicates, and which are not. While this may seem like a straightforward task — just find the verb as in a simple sentence like "He *ate* the apple." —, there are actually a few caveats (predicates are emphasised): (1) There may be no predicates at all: "What a day!". (2) There might be more than one predicate: "We *saw* her *leave* the room". (3) Not all verbs might be predicates: "I can *hear* you". In the following section, I will describe how I tackle these problems by making use of parsing information from ParZu.

germanBERT

Subtoken-Fusion-Module

Self-Attention

| [CLS] | Wir | wussten | nicht | , | was | eine | U | ##2 | war |

Wir wussten nicht, was eine U2 war ...

ParZu

Wir wussten nicht , was eine U2 war

subj · adv · det · pred · subj · objc

DAMESRL

| [CLS] | B-A0 | B-V | O | O | B-A1 | I-A1 | I-A1 | I-A1 |

GRU-Layer

Split-SRL-module

HEAD
- [CLS]
- FFNN
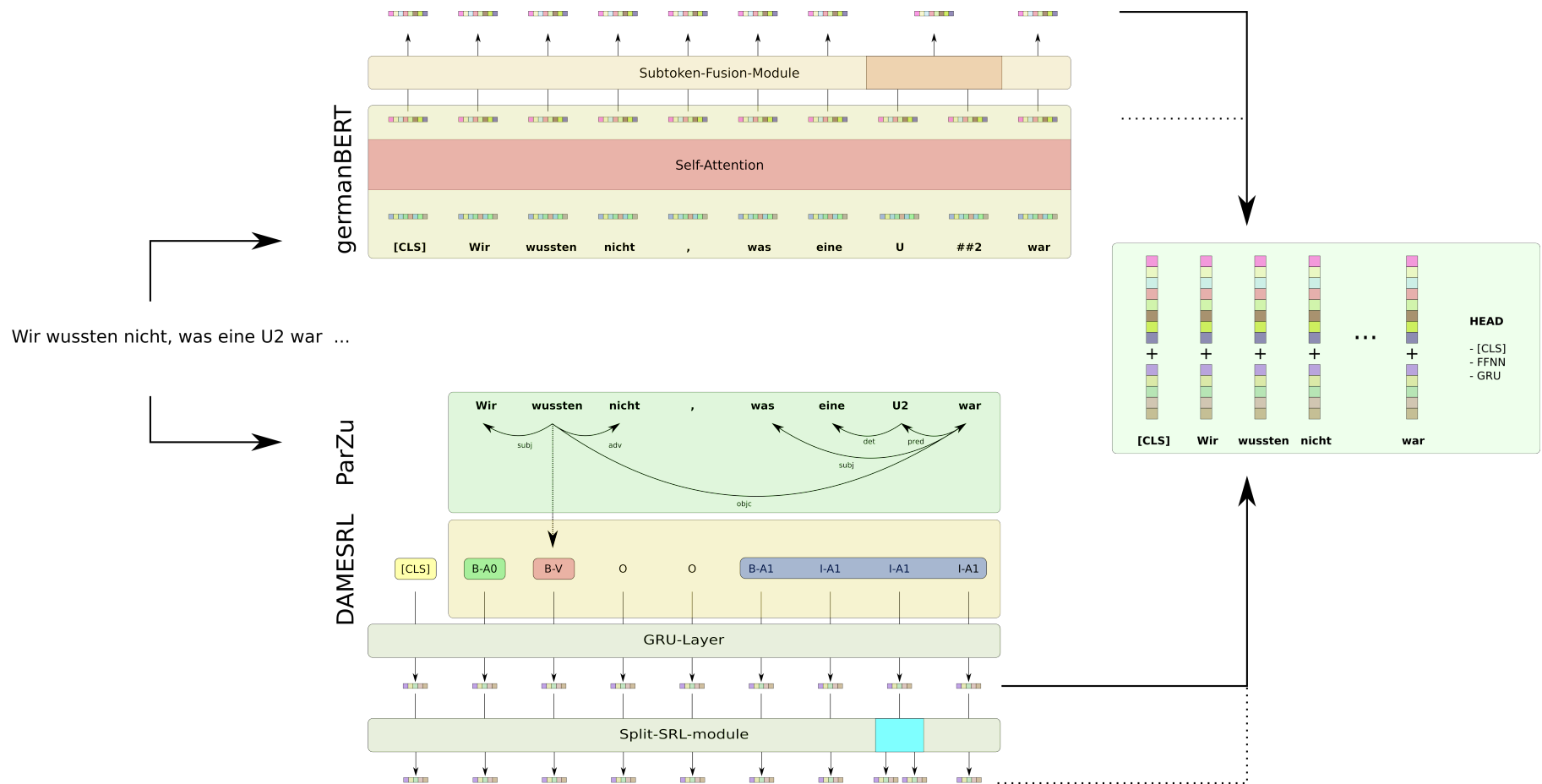- GRU

| [CLS] | Wir | wussten | nicht | war |

Figure 12: Detailed architecture of gliBERT.

4.3.1 Finding Predicates

It is a known problem in the analysis of semantic roles that a proper procedure for predicate identification is a problem hard to tackle, consider e.g. the discussion concerning so called light verbs: Wittenberg [2016].

"First, the predicates which assign semantic roles to the constituents are identified prior to semantic role labelling proper. They are usually identified as the main verbs which head clauses." [Samardzic, 2013, p. 74] In a dependency framework like the Universal Stanford Dependencies (USD) [De Marneffe et al., 2014], which explicitly sets the content verb as root, identification of the relevant predicate is straightforward: One has simply to look at the dependency parse tree of a given sentence and select the verbal heads — i.e. roots — of the clauses. However, the ParZu parser models not content verbs as heads but function verbs.[2]

(interestingly, this stands in contrast to the Pro3Gres parser [Schneider, 2008] which

"In a constituency parse, the finite verb is the head of a verb phrase or rather sentence. A dependency parse, on the other hand, does not consider auxiliaries as heads and therefore finite verbs are usually not the head of the sentence. Hence, the head of a sentence typically is the verb containing the meaning. In that sense, dependency structures are closer to the semantics of a sentence." [Aepli, 2018, p. 6f.]

According to the USD, function words are subordinated to content words, which means that in a sentence "He was hit by a ball.", the infinite participle *hit* would be analysed as root, not the finitely inflected *was*. This is an accordance with the view that XXXXXXXXXX However, there is a "substantial amount of evidence [that] delivers a strong argument for the [...] approach, which subordinates full verbs to auxiliaries" Groß and Osborne [2015].

"The parsing scheme that USD advocates takes the division between function word and content word as its guiding principle. One major difficulty with doing this is that the dividing line between function word and content word is often not clear." Groß and Osborne [2015]

Following Foth [2006]

(4.1) Die Keita-Dynastie regierte das vorkaiserliche und kaiserliche Mali vom 12. Jahrhundert bis Anfang des 17. Jahrhunderts.

(4.2) Im tibetischen Buddhismus werden die Dharma-Lehrer/innen gewöhnlich als

---

[2]This follows general dependency frameworks proposed for German, e.g. Gerdes and Kahane [2001]; Groß and Osborne [2015].

Lama bezeichnet.

(4.3) Die Klage wurde abgewiesen, was als Sieg beschrieben werden kann.

whose dependency parse tree is shown in Figure 13: This sentence has five verbs in it, *wurde*, *abgewiesen*, *beschrieben*, *werden*, and *kann* (POS-tag "V" in the second row), but only two of them are relevant predicates, i.e. predicates that carry "true" semantics.
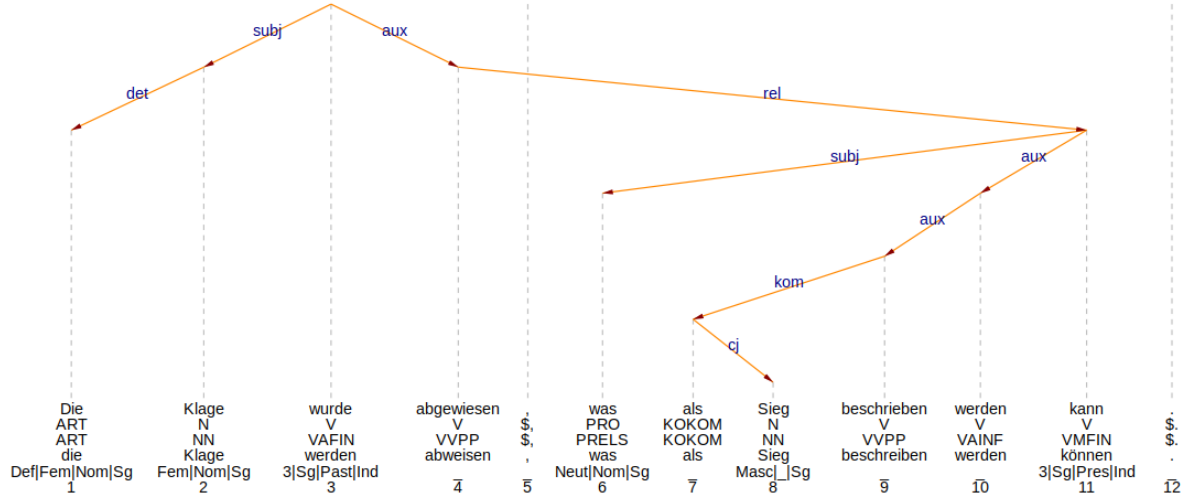
Figure 13: Example dependency parse tree for a sentence with multiple predicates.

I propose the following algorithm 1 deciding whether a verb in a sentence is or is not a predicate using a heuristic, relying on the token's POS tag that the parser predicts. The ParZu parser's default output follows the CoNLL scheme [Buchholz and Marsi, 2006] which means that there are two levels of POS tagging: coarse-grained (CPOSTAG) and fine-grained (POSTAG), where the POSTAG corresponds to the token's STTS tag [Schiller et al., 1999].

The condition on line 9, that only tokens in the respective subclause are considered, is ensured by making sure that if a token u's POS is "V" and it points to its head t, that it is not itself the head of a subclause — i.e. its dependency relation is e.g. "relative clause". If that is the case the token u is considered to belong to another subclause and therefore not preventing token t from getting labelled as a predicate. Consider again the example 4.3.1: Let's say we are in the for-loop at the token *weitergeleitet*. Because it is a verb but not a finite full-verb, we enter the else-clause on line 7. If we were now to loop through all token of sentence 4.3.1 we would find that token *führt* is a verb that points to our primary token. Without the above outlined constraint that only verbs in the same subclause pointing to our original verb are preventing it from being labelled a predicate, *weitergeleitet* would

---

Algorithm 1 Predicate finding algorithm

1: **for all** token $t \in$ sentence **do**
2:     **if** CPOSTAG $t \neq$ 'V' **then**
3:         $t \leftarrow$ NOT_PRED
4:     **else**
5:         **if** POSTAG $t =$ 'VVFIN' **then**
6:             $t \leftarrow$ PRED
7:         **else**
8:             FLAG $\leftarrow True$
9:             **for all** token $u \neq t \in$ subclause **where** $t \in$ subclause **do**
10:                **if** CPOSTAG $u =$ 'V' $\wedge\ u$ dependent on t **then**
11:                    $t \leftarrow$ NOT_PRED
12:                    FLAG $\leftarrow False$
13:                    **break**
14:                **end if**
15:            **end for**
16:            **if** FLAG $= True$ **then**
17:                $t \leftarrow$ PRED
18:            **end if**
19:        **end if**
20:    **end if**
21: **end for**

---

be labelled as non-predicate. This is obviously false. Taking into account the above considerations, we see that although *führt* points to *weitergeleitet*, its edge label is *rel* — which means that it's the head of a relative subclause — therefore it is not anymore in the same subclause and *weitergeleitet* gets labelled as predicate.

4.3.2 Ensuring Tokenization Equivalence

One of the major difficulties I ran into, was the tokenization differences between different parsers. In conrete terms, this means that it is not always possible to correctly align the tokens which two parsers produce for the same sequence. The DAMESRL system implements the tokenizer provided by the Natural Language Toolkit (NLTK)[3] which implements a linguistically motivated tokenizing. explain what that means BERT, in contrast, utilizes an approach called "WordPieces", which is a rather information processing motivated approach: "Using wordpieces gives a good balance between the flexibility of single characters and the efficiency of full words for decoding, and also sidesteps the need for special treatment of

---

[3]https://www.nltk.org/

unknown words." [Wu et al., 2016, p. 2]. Although the NLTK algorithm is guided by linguistically informed rules and statistic while the WordPieces approach simply reflects distribution properties of assembled letters, both systems tokenize sentences in the same way in most cases. However, especially when rare symbols such as currencies, units, and the like are present, the tokenization slightly differs What is even worse, often the correct alignment of those differing sequences is rather complicated to obtain. automatically

To illustrate this, consider the following sentence from the PAWS-X data set:

(4.4) Die mittlere Oberflächentemperatur wird auf -222 °C (~51 K) geschätzt.

BERT (merged)	**NLTK**
Die	Die
mittlere	mittlere
Oberflächentemperatur	Oberflächentemperatur
wird	wird
auf	auf
-	-222
222	°C
[UNK]	(
(	~51
~	K
51	)
K	geschätzt
)	.
geschätzt	
.	

The first question that arises is: which tokenization should be mapped onto wich? In other words: should we try to align the BERT tokens with the corresponding NLTK tokens or vice versa? Let's assume we decide to align the tokenization T with fewer items to the one with more items — in this case this would mean aligning T_{NLTK} with T_{BERT}. So, the first five tokens are no problem, we can align them by simply doing an exact match and confirm that the elements correspond. But when we reach the sixth token, the exact match fails. To decide whether the token $t_{T_{BERT}}$ or the token $t_{T_{NLTK}}$ was split up — i.e. to determine which token must be copied to ensure tokenization equality —, we need to do a mutual substring match. Doing this, we eoudl find out that "-" is a substring of "-222". In consequence, we align the two, duplicate "-222" and compare it with token number 7 in T_{BERT}. Since "222" is a substring of "-222", so we align the two of them.

While it is theoretically possible to align tokens that were differently tokenized by the two algorithms, it is nevertheless quite cumbersome. The main problem, however, arises due to the [UNK] token BERT introduces for characters — or character sequences — which lie out of its vocabulary. Since there is obviously no more (sub-)string comparison possible, the process gets even more complicated: Suppose you have duplicted the "-222" in the NLTK column and are now on line 7. In the BERT tokenization you see the "[UNK]" token, while in the NLTK you see a "°C". To find out, what all is containt in the "[UNK]", you need to look at the token before and after it in the BERT tokenization and compare it with the respective NLTK tokens. since the the and so on.... the BERT-tokenized sequences, to get around this issue.

BERT (merged)	**NLTK**
Die	Die
mittlere	mittlere
Oberflächentemperatur	Oberflächentemperatur
wird	wird
auf	auf
-	-222
222	-222
[UNK]	°C
(	(
~	~51
51	~51
K	K
)	)
geschätzt	geschätzt
.	.

4.3.3 DAMESRL

There are not too many SOTA SRL frameworks available for German that come with a pre-trained model, especially such ones that can be integrated in a pipeline in a pipeline of a bigger system.

Do et al. [2018] fill exactly this hole: They introduce DAMESRL, an SRL framework that implements SOTA architecture, namely self-attention mechanisms, similar to BERT's. They report an F1 score of 73.5 for their best model configuration on the German data set of CoNLL '09. This best configuration is based on word as well as character embeddings, self-attention and a softmax layer on top.

The DAMSRL predictor receives the BERT-tokenized sentence along with the information which tokens in it are predicates (zero or more). For each token labelled as predicate in a sequence it predicts for each other token in the sequence its SRL.

4.3.4 GRU

Finally, the predicted SRLs need to be encoded in a numeric way similar to the BERT embeddings so . that they can be combined and sent thorugh the final head network Encoding a sequence is a typical. seq2seq task, for which recurrent neural networks have proven to be effective mechanisms (CITE) .

- number of predicates equals 3

- duplicate or zeros when not enough predicates

- concatenate all sentences and then encode vs. encode each sentence and then concat

- ...

- Embed each sentence alone vs. concatenate all sentences

- same for text_1, text_2

- add meta-SRLs [CLS], [SEP]

- duplicate if less preds than 3 vs. fill with zeros

4.4 combination

4.4.1 Aligning BERT subtokens with SRL tokens

A crucial part in the overall architecture is the combining of the numeric representation of (sub-)words computed by the BERT network and the embedded SRLs. One difficulty lies in the fact that, as already mentioned above, BERT has its own tokenizer which implements a so-called sub-word or wordpiece Wu et al. [2016] encoding strategy: BERT has a fixed length of vocabulary it can hold, namely 30,000 tokens. The wordpiece tokenization approach is a balance between word and character tokenization in that that it "gives a good balance between theflexibility of single characters and the efficiency of full words for decoding, and also sidesteps the need for special treatment of unknown words" [Wu et al., 2016, p. 2]. As a

consequence, BERT tokenizes a sentence quite differently than a traditional parser, since the latter adheres to the full tokens. Consider the following example:

(4.5) Es ist der Sitz des Bezirks Zerendi in der Region Akmola.

(4.6) ParZu: Es ist der Sitz des Bezirks Zerendi in der Region Akmola .

(4.7) germanBERT: Es ist der Sitz des Bezirks Zer ##end ##i in der Region Ak ##mol ##a .

A further challenge besides the alignment of traditional tokenization and wordpiece tokenization is the general difference in parsing a sentence that exist.

4.5 Head Module

4.5.1 Classifiction

While for question answering there was little tweeking needed to adapt to the extended BERT embeddings, for classification the situation looks a bit more complex. The standard BERT way of doing classification tasks runs as follows:

- Prepare the data: add a [CLS] token at the beginning, a [SEP] token between the two sentences (if there are), and pad with the [PAD] token

- Send the prepared examples through the BERT network

- Select only the embedding for the first token — i.e. the [CLS] —, send it through a dense layer with a softmax and predict the class for this example

Devlin et al. [2018] visualize this as can be seen in figure 14.

The problem now is that in the above described standard implementation, there is no straigthforward way to enrich the BERT embeddings with SRLs, since the only embedding that is used for prediction is the [CLS] token; since this is a special BERT token it is not present in the original sentence and, therefore, it does not have a corresponding SRL. (And what should that be? Since it is a meta-token it rcan't really have a SRL?

To tackle this problem, I remodeled the architecture from Zhang et al. [2019b] on the one hand, and on the other hand tested several other final layer designs.
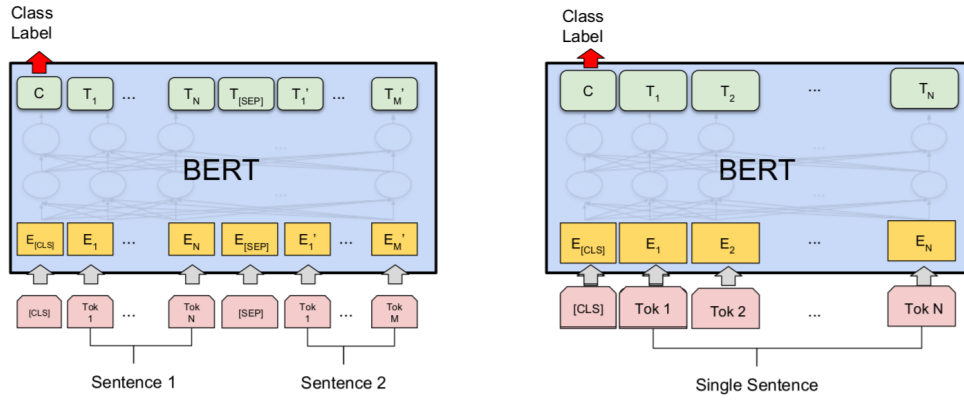
Figure 14: Schema for sentence pair (left) and single sentence (right) classification. Figure taken from [Devlin et al., 2018].

4.5.1.1 [CLS] Head

In their paper for SemBERT, Zhang et al. [2019b] do not really address the issue laid out above. To the contrary, the differnt pieces of information they provide are rather conflicting, only after inspecting the code they released on GitHub, the picture somewhat cleared:

After predicitng the SRLs for a given input, they add pseudo-SRLs for the [CLS] and [SEP] tokens. In the look-up table of the BiGRU that consumes the SRLs, they then simply add the corresponding keys — so that besides regular SRLs as "B-V" (beginning of predicate) or "I-A0" (inside or end of argument zero), there are also the labels "[CLS]" and "[PRED]" After sending this sequence through the BiGRU, they concatenate the two hidden states of the [CLS] SRL with the [CLS] BERT embedding and predict on that vector SRL for the [CLS] token after the sequence was sent through the BiGRU which then can be appended to its BERT embedding vector

In table XYZ I report the gains, losses this strategy yields for the four classifications data sets in my corpus.

4.5.1.2 FFNN Head

FFNN stands for Feed Forward Neural Network. While th [CLS] Head procudes predictions based on the weights of the last layer output of the [CLS]-token, this head takes the last layer output of all tokens and concatenates them. While the approach implemented by Zhang et al. [2019b] is able to improve the vanilla BERT approach,
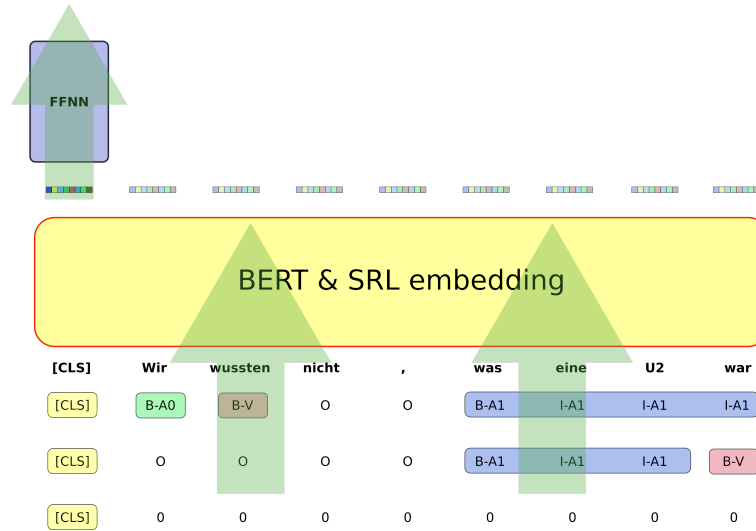
Figure 15: Head with a one-layer **F**eed **F**orward **N**eural **N**etwork on the [CLS]-token

it does not lead to an improvement on others, or worse, brings the perormance down. I suspect a reason for this may lie in the manner of how the SRLs are processed in this approach. The information SRLs provide is, what may be called, sub-sentence specific and cannot be adequately represented as a single information piece. By this I mean that it does not suffice to know that given an utterance x that there is a specific SR in it; rather the information *where* is crucial. Consider the following example (the pseudo SRL [CLS] is added):

[[CLS]] [A-0 The man] [predicate asked] [A-1 his friend] .

After the subscripted SRLs were consumed by the BiGRU, there is some information about all SLRs in the hidden states of the [CLS] token. While there may be some information about there being a predicate, an argument zero, and an argument 1 present, it is completely impossibly to determine from which tokens these signals came. Especially in sentence pair tasks, such as paraphrase identification, this information is however absolutely crucial. As can be seen from results 7, this hypothesis is also supported by the results:

As has been shown by e.g. Myagmar et al. [2019] for sentiment analysis, a simply final fully-connected feed forward layer produces fairly good results (in fact, it performs the best in the different architectures they tested for their task).

Implementing a fully-connected feed forward network as final layer counters the problem of the information deprecation that is present in the [CLS] approach: Every token's BERT embedding gets concatenated with the token's SRL embeddding. The
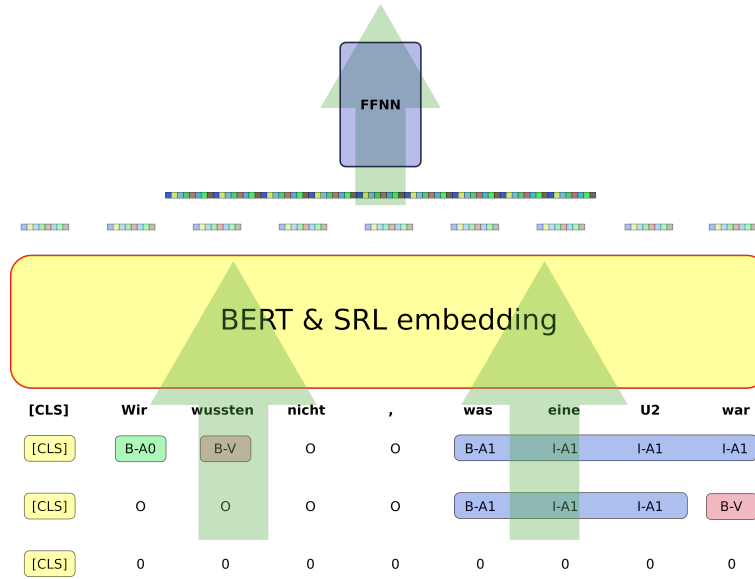
Figure 16: Head with a one-layer **F**eed **F**orward **N**eural **N**etwork on the concatenated token sequence.

whole sequence is then flattened, i.e. all BERT+SRL vectors get concatenated into one large vector of size $\mathbb{R}^{n \times 768 + XXX}$

4.5.1.3 GRU Head

Since the SRLs are essentially a sequential "mark-up" of the sentences, the thought of encoding them with an architecture designed for sequential data is not too far. Inspired by the biological properties of the brain, the concept of recurrent neural networks has been around since the late 80ies, with [Hopfield, 1982] often being credited as having implemented the first recurrent neural network. To overcome the problems of the vanishing and explofing gradient problems, [Hochreiter and Schmidhuber, 1997] proposed the LSTM architecture. [Cho et al., 2014] GRU

4.5.2 Question Answering

4.5.2.1 Span Prediction Head
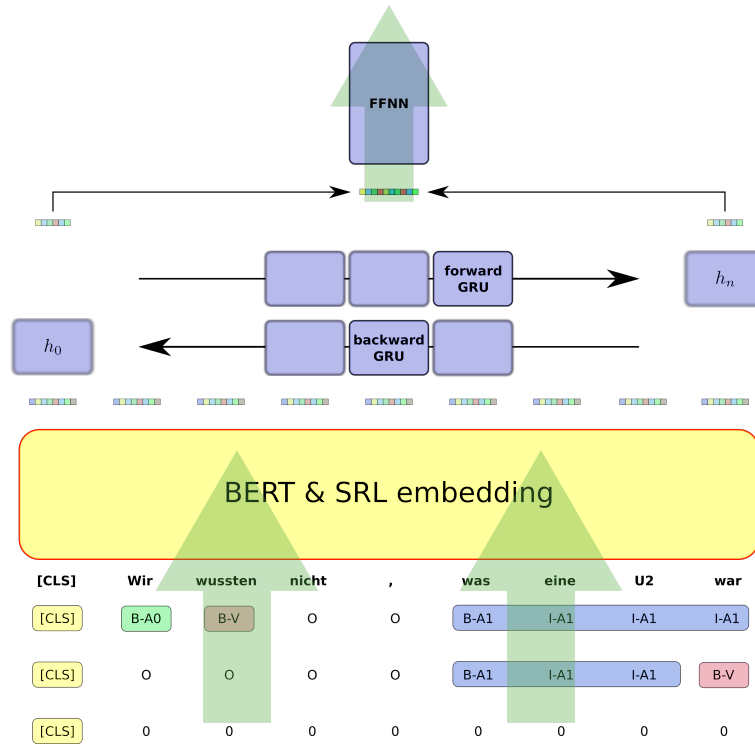
Figure 17: Head with a one-layer **F**eed **F**orward **N**eural **N**etwork on the concatenated last hidden states of the bi-directional GRU.
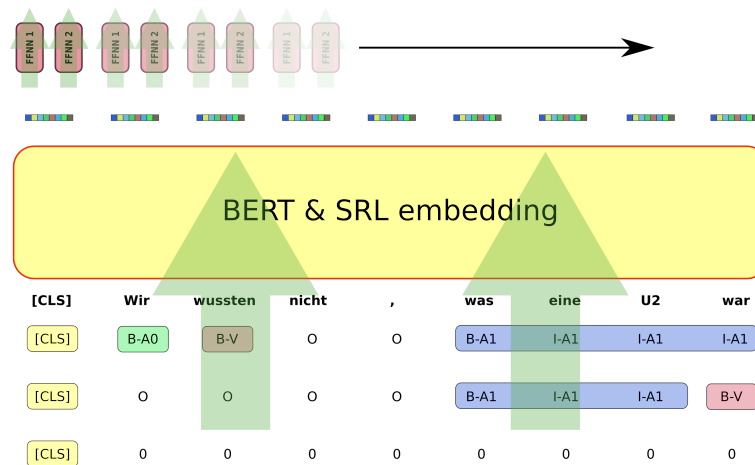
Figure 18: The gliBERT head for span prediction in Question Answering Tasks. After the Tokens and SRLs were consumed by BERT and the SRL embedding module, two FFNNs predict on each token in the sequence, how likely it is the start or the end of the answer span. After these predictions have been made for all tokens, the token with the highest value for each position get selected.

5 Results

In this chapter, I will report the results of my experiments on the six data sets. Additionally, I conducted an ablation study on the XNLI data set, that is described in section 5.6

5.1 Data Set Results

- conjecture: SRLs are rather adding noise in sequences that are too long. Extreme exmples are the Q&A datasets.

- GRU architecture is probably strongest: most best models (even though mostly -SRL) and second best models, no worst performance

- 6 sgnificantly better +SRL vs. 3 significantly worse +SRL. There seems to be a slight trend that when merging subtokens, duplicating SRLs when too less predicates is better.

- all 3 worsening are for subtokenized architectures.

To obtain as stable results as possible, I decided to train five models for each architecture and configuration, all initialized with different random seeds. Additionally, I ensembled the five models, achieving a performance gain of several percentage points (see example of PAWS-X, table **??**). In table 7 below, the test set ensemble results for each architecture on the non-question answering data set are reported. The results for the architecture for the question answering tasks are reported in table 14.

In a first step, I will discuss on the overall performance of models when the SRLs are added, compared to the same architectures without.

Classification Data Sets

		[CLS] Head						FFNN Head						GRU Head					
		subtokenized			subtokens merged			subtokenized			subtokens merged			subtokenized			subtokens merged		
		−SRL	+SRL		−SRL	+SRL		−SRL	+SRL		−SRL	+SRL		−SRL	+SRL		−SRL	+SRL	
			zeros	dupl.		zeros	dupl.		zeros	dupl.		zeros	dupl.		zeros	dupl.		zeros	dupl.
deISEAR	α	71.52	72.19	71.52	72.19	*67.55*	72.19	70.86	**77.48**	72.85	74.17	72.85	74.17	70.20	74.83	74.17	73.51	70.20	71.52
	β	71.52	74.83	74.83	70.86	70.86	72.85	74.83	*68.21*	70.20	74.83	73.51	70.86	73.51	74.83	72.19	**76.82**	70.20	74.83
SCARE	α	85.61	82.58	83.71	83.33	83.71	85.61	83.33	83.71	84.09	84.09	*81.44*	84.47	83.71	83.33	84.09	**85.98**	84.09	83.71
	β	86.36	84.85	84.47	85.23	85.23	85.23	**86.74**	85.98	85.23	84.47	*83.33*	84.09	**86.74**	85.98	83.71	86.36	84.09	85.23
PAWS-X	α	80.63	81.60	81.49	*79.92*	80.63	82.51	81.19	80.78	80.07	80.43	80.02	80.68	82.26	82.77	82.77	82.82	82.87	**83.53**
	β	87.49	88.05	**88.21**	87.75	87.24	88.00	86.83	87.39	87.09	87.75	*86.58*	86.68	87.60	87.60	87.90	88.00	88.00	88.05
XNLI	α	67.34	**67.52**	66.64	66.94	66.94	*66.26*	67.20	67.42	67.34	66.38	67.08	66.92	66.68	66.60	67.14	66.42	66.54	66.26
	β	68.09	68.18	66.84	67.82	67.82	68.36	66.31	65.60	66.40	*64.98*	65.51	65.07	66.84	67.82	67.02	67.64	66.31	**68.53**
Scores		II	**II** II			II		**I** I	I I		I			**I**	II		**II** I		**II** III
+SRL		**5** 10																	
−SRL		**4** 5																	

Table 7: Accuracy ensemble results (per 5 models) on single sentence and sentence pair tasks. **Bold** font marks the best result per line, underline the second best, and *italics* the poorest. In the *Scores* row, the afore mentioned positive extremes are accumulated for −SRL and +SRL; note that if both +SRL configurations of an architecture achieved an extreme, it is only counted once.
The line marked with light gray — PAWS-X β — is "expanded" in table 8 to illustrate that each results in this table is actually a majority voting out of an ensembeling of fice models.

PAWS-X β

	[CLS] Head						FFNN Head						GRU Head					
	subtokenized			subtokens merged			subtokenized			subtokens merged			subtokenized			subtokens merged		
	−SRL	+SRL		−SRL	+SRL		−SRL	+SRL		−SRL	+SRL		−SRL	+SRL		−SRL	+SRL	
		zeros	dupl.		zeros	dupl.		zeros	dupl.		zeros	dupl.		zeros	dupl.		zeros	dupl.
Model 1	85.41	85.36	86.02	86.43	86.53	87.04	85.77	85.61	85.77	86.22	84.70	86.53	87.54	87.49	86.43	85.51	86.93	86.53
Model 2	86.07	86.83	86.99	85.87	86.32	86.68	86.17	85.82	87.39	85.92	85.36	85.26	87.04	86.99	85.87	87.19	86.07	87.44
Model 3	86.07	87.49	86.99	87.14	85.26	86.73	86.22	84.90	85.36	85.41	85.31	85.71	86.12	85.66	86.83	86.48	87.09	86.93
Model 4	87.39	86.32	86.58	86.38	84.04	87.65	86.53	86.73	85.61	85.82	85.61	86.38	84.99	86.02	87.70	86.99	86.68	86.88
Model 5	86.63	86.43	86.58	86.99	85.26	85.56	86.18	87.09	84.75	87.04	85.77	85.82	86.12	85.82	86.73	87.34	86.73	86.58
Average	86.31	86.49	86.63	86.56	85.48	_86.81_	86.22	86.03	_85.78_	86.08	85.35	85.94	86.35	86.40	86.71	86.70	86.70	**86.87**
Ensemble	87.49	88.05	**88.21**	87.75	87.24	88.00	86.83	87.39	87.09	87.75	_86.58_	86.68	87.60	87.60	87.90	88.00	88.00	88.05
Gain	1.18	1.56	1.58	1.19	1.76	1.19	.65	1.36	1.31	1.67	1.23	.74	1.25	1.20	1.19	1.30	1.30	1.18
Average	1.26 (σ 0.28)																	

Table 8: The "expanded" PAWS-X β results. The light gray line corresponds to the one in table **??**. As can be seen, the fluctuations between single models is not too big, which is an indicator that the architecture is fairly stable. Ensembeling reliably adds 1.26 percentage points on average.

		[CLS] Head		FFNN Head		GRU Head	
		zeros	dupl.	zeros	dupl.	zeros	dupl.
deISEAR	α	4.64**	.67	4.63*	1.32	4.63*	2.65
	β	3.97*	1.98	5.30*	.66	4.63*	2.64
SCARE	α	1.13	1.90	2.27*	.38	.76	.38
	β	.38	.76	2.65**	1.14	1.89	1.52
PAWS-X	α	.97*	1.02**	.76	.61	.10	.76
	β	.81**	.21	.81*	.41	.40	.15
XNLI	α	.58**	.38	.34	.42	.06	.88**
	β	.36	1.52*	.09	1.33*	1.51*	1.51*

Table 9: Performance of architectures when BERT subtokenized (yellow) vs. merged (purple). Both SRL implementations were compared pairwise. Example case, upper left 4.64**: Comparison of deISEAR α, +SRL zero implementation, [CLS] Head, subtokenized ensemble (72.19% accuracy) and the merged ensemble (67.55% accuracy) — the subtokenized ensemble performed 4.64% better, apparently with very high significance ($p < 1\%$).

		# of epochs	split set up	batch size	maximum length	SRL implementation
deISEAR	α	100	normal	16	40	normal
	β	100	normal	16	200	normal
SCARE	α	50	normal	16	50	normal
	β	50	normal	16	100	normal
PAWS-X	α	20	normal	16	16	normal
	β	20	normal	16	16	normal
XNLI	α	50	normal	16	100	normal
	β	50	re-split	16	100	normal
MLQA	α	40	200	50	100	normal
	β	40	200	50	100	normal
XQuAD	α	40	200	50	100	100
	β	40	200	50	100	100

Table 10: The different hyperparameter configurations for each data set.

5.1.1 Testing for Statistical Significance

"if we rely on empirical evaluation to validate our hypotheses and reveal the correct language processing mechanisms, we better be sure that our results are not coincidental." [Dror et al., 2018]

$$\delta(X) = M(A, X) - M(B, X)$$

$$H_0 : \delta(X) \leq 0$$

$$H_1 : \delta(X) > 0$$

learning rate	2e-05
SRL embedding dimensions	20
SRL GRU hidden size	32
SRL number of layers	2
SRL bias	True
SRL bidirectional	True
SRL dropout	0.1

Table 11: General hyperparameter configurations.

		[CLS] Head		FFNN Head		GRU Head	
		subtok.	merged	subtok.	merged	subtok.	merged
deISEAR	α	.67	.00	6.62**	.00	4.63**	-1.99
	β	3.31	1.99	$-4.63*$	-1.32	1.32	-1.99
SCARE	α	$-1.90*$	2.28*	.76	.38	.38	-1.89
	β	$-1.51**$	.00	$-.76$	$-.38$	.76	-1.13
PAWS-X	α	.97*	2.59***	$-.41$	.25	.51	.71
	β	.72*	.25	.56	$-1.07***$	.30	.05
XNLI	α	.18	$-.22$	.22	.70*	.46	.12
	β	.09	.44	.09	.53	.98	.89

Table 12: Ensemble percentage points gains (positive numbers) / losses (negative numbers) for +SRL over −SRL for each configuration from table 7. The better of the +SRL configurations was taken into account: zeros , duplicate . Light blue denotes that both architectures performed equally (in which case both ensembles were controlled for significance). One asterisk signifies a p-value $< 10\%$, two stand for $p < 5\%$ and three for $p < 1\%$.

"It is important to have a method at hand that gives us assurances that the observed increase in the test score on a test set reflects true improvement in system quality." [Koehn, 2004]

Koehn [2004] focus strongly on significance testing in the context of evaluating on a sub-sample of the test set — due to expensiveness of testing on the whole set — and making statements about the reliability of this subset sample:

"Given a test result of m BLEU, we want to compute with a confidence q (or p-level P = 1 - q) that the rue BLEU score lies in an interval $[m - d, m + d]$." [Koehn, 2004]

Since the systems under review here predict on the exact same test set, the assumed independence of the predictions of the two models holds no longer. Morgan [2005]
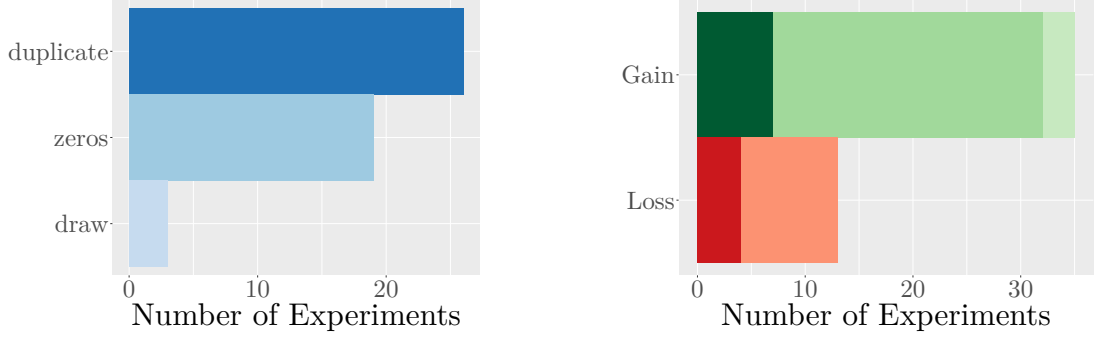
Figure 19: Accumulated scores from table 12. **Left**: Which SRL mode performed stronger out of a total of 48 settings. The bars indicate a slight outperforming of duplicating SRLs instead of adding zeros. **Right**: Counts for Gains and Losses in all 48 settings. Darker shades indicate significant results. The light green tip stands for settings where the gain equals 0.00.

		Predicted		
		Positive	Neutral	Negative
	Positive	149	12	5
True	Neutral	4	11	6
	Negative	3	8	66

Table 13: Confusion matrix for SCARE α merged, +SRL duplicated. (macro F1: 73.52).

propose the following algorithm for testing difference significance:

"When the results are better with the new technique, a question arises as to whether these result differences are due to the new technique actually being better or just due to chance. Unfortunately, one usually cannot directly answer the question "what is the probability that the new technique is better given the results on the test data set"" [Yeh, 2000]

"But with statistics, one can answer the following proxy question: if the new technique was actually no different than the old technique (the null hypothesis), what is the probability that the results on the test set would be at least this skewed in the new technique's favor?" [Yeh, 2000]

Many evaluation metrics "have a tendency to underestimate the significance of the results", due to their inherent assumption that the compared systems "produce independent results" when in reality, they tend to produce "positively correlated results". [Yeh, 2000]

Algorithm 2 Approximate Randomization Algorithm

1: $p(M, x)$ = prediction of model M on example x
2: A, B = Two different models
3: $O = \{x_1, \ldots, x_n\}$ = test set
4: $O_A = \{p(A, x_1), \ldots, p(A, x_n)\}$
5: $O_B = \{p(B, x_1), \ldots, p(B, x_n)\}$
6: O_{gold} = gold labels for $\{x_1, \ldots, x_n\}$
7: $e(\hat{Y}, Y)$ = evaluation function for gold labels $\hat{Y}$ and predictions Y
8: $t_{original} = \mid e(O_{gold}, O_A) - e(O_{gold}, O_B) \mid$
9: $rand()$ = returns 0 or 1, randomly
10: $swap(x, y)$ = exchanges elements $x \in A, y \in B$ such that $y \in A, x \in B$
11: $r \leftarrow 0$
12: $R \leftarrow 0$
13: $threshold \leftarrow 1,000$
14: $p \leftarrow 0.05$
15: **while** $R < threshold$ **do**
16:     **for all** $(a_i, b_i) \in O_A \times O_B \mid i \in I$ **do**
17:         **if** $rand() = 0$ **then**
18:            $swap(a_i, b_i)$
19:         **end if**
20:     **end for**
21:     $t_{permute} = \mid e(O_{gold}, O'_A) - e(O_{gold}, O'_B) \mid$
22:     **if** $t_{permute} \geq t_{original}$ **then**
23:         $r \mathrel{+}= 1$
24:     **end if**
25:     $R \mathrel{+}= 1$
26: **end while**
27: **if** $\frac{r+1}{R+1} < p$ **then**
28:     system A truly better than system B
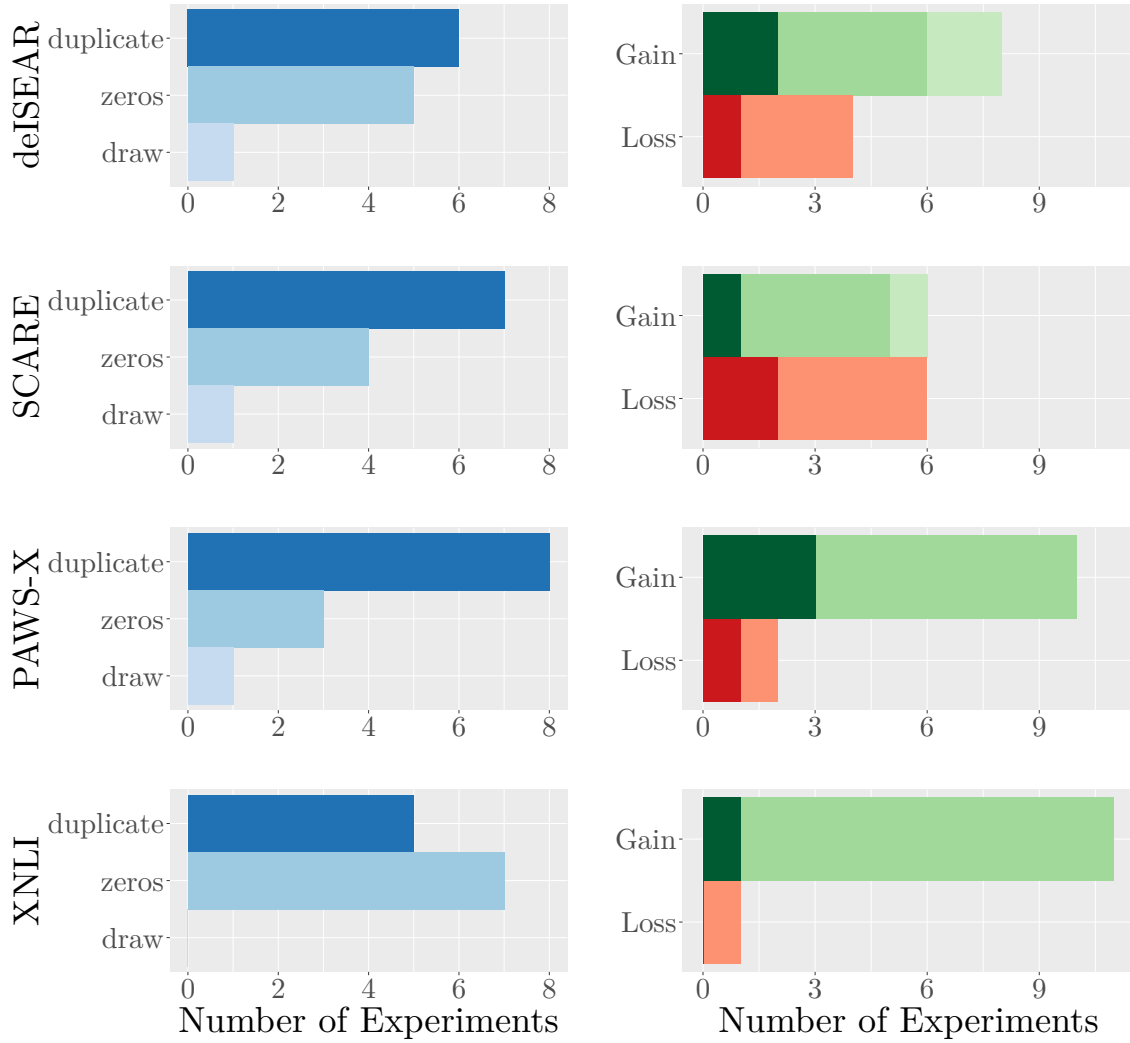29: **end if**

Figure 20: Accumulation of statistics for each classification datasetclassification. **Left**: Which SRL architecture performed better. **Right**: Comparison of accuray points gained/lost after adding SRLs.

5.1.1.1 Example Case for XNLI

Let's consider the case for the non-merged subtokens setting in the resampled XNLI data set. The test set contains 1,125 sentence pairs for which textual entailment must be predicted. From these 1,125 sentence pairs, 398 bear the gold label *contradiction*, 357 are labeled *entailment*, and 370 are *neutral*; so, the class distribution of the set is fairly balanced.

I trained and optimized five systems for two architectures on the training and development set of XNLI: One architecture is the plain "vanilla" GRU classifier described in section XXX, the other is the same GRU architecture enriched with em-

Q&A Data Sets

Span Prediction Head

		subtokenized			subtokens merged		
		−SRL	+SRL		−SRL	+SRL	
			zeros	dupl.		zeros	dupl.
MLQA	α	**30.69**	<u>29.68</u>	29.68	21.92	21.92	*21.81*
	β	**44.75**	<u>44.55</u>	43.41	*41.66*	41.86	41.79
XQuAD	α	**42.01**	<u>41.42</u>	41.12	37.87	36.98	*35.50*
	β	<u>46.57</u>	45.43	**46.86**	37.43	*37.14*	39.14
Scores		**III** <u>I</u>	**I** <u>III</u>				
+SRL	**1** <u>3</u>						
−SRL	**3** <u>1</u>						

Table 14: Accuracy ensemble results (per 5 models) on question answering tasks. **Bold** font marks the best result per line, <u>underline</u> the second best, and *italics* the poorest.

bedded SRLs (implementing the duplication approach, described in section XXX). The "vanilla" system ensemble achieved an accuracy of 66,58% on the XNLI test set, while the SRL enriched ensemble scored a 68,27% — in other words, the SRL enchriched ensemble performed 1,69% better than the "vanilla" ensemble.

To check if this difference truly measures the supremacy of the latter model over the first, I apply the above described algorithm 2 for testing significance by permuting the actual ensemble predictions. Note that both ensemble models were equally right or wrong in 1,018 cases out of 1,125. From this follows, in consequence, that in 90,49% of the cases the flipping of predictions between the ensemble models will have no effect.

Result p = 4.80%

In contrary, if we compare this results to the zero implementation of SRLs, we observe something different: The accuracy of this ensemble was slightly lower than the duplicate architecture; namely 67,73% or, speaking in differences, 1.15% better than the vanilly ensemble. The number of equally right or wrong examples was also slightly lower — 1,010 examples were equally wrong or correct predicted by the systems.

Result zeros p = 14.09%

Span Prediction Head

		Gains/Losses		subtokenized/merged	
		subtok.	merged	zeros	dupl.
MLQA	α	−1.01***	.00	7.76***	7.86***
	β	−.20	.20	2.69	1.62
XQuAD	α	−.59	-.89	4.44**	5.62***
	β	.29	1.71	8.29***	7.72***

Table 15: Left part: Ensemble percentage points gains (positive numbers) / losses (negative numbers) for +SRL over −SRL for the Span Prediction Head from table 14. The better of the +SRL configurations was taken into account: zeros , duplicate . Light blue denotes that both architectures performed equally (in which case both ensembles were controlled for significance). One asterisk signifies a p-value $< 10\%$, two stand for $p < 5\%$ and three for $p < 1\%$. **Right part**: asdasdasdasdasd.
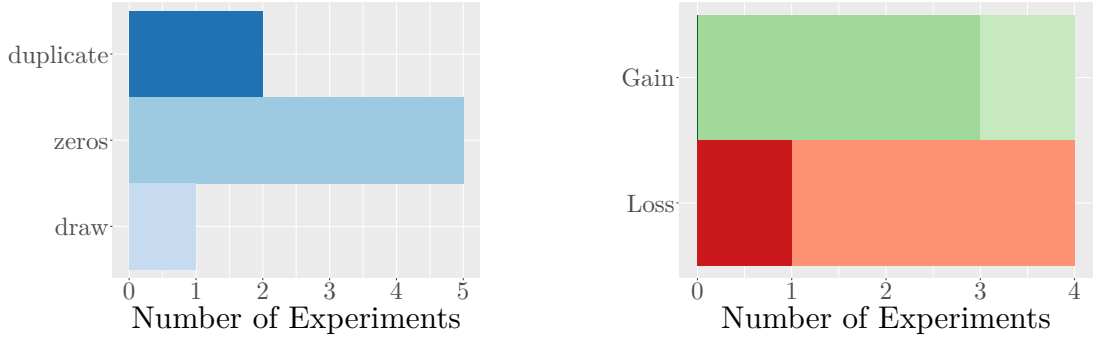

Figure 21: Accumulated scores from table **??**. **Left**: Which SRL mode performed stronger out of a total of 8 settings. The bars indicate a slight outperforming of duplicating SRLs instead of adding zeros. **Right**: Counts for Gains and Losses in all 8 settings. Darker shades indicate significant results. The light green tip stands for settings where the gain equals 0.00.

In summary it is safe to say that although there is a positive effect of injecting SRL information during training over all data sets and architectures, this effect is arguably quite small and unsteady. this is especially in yontrast to Zhang et al. [2019b], who report more stable and higher effects In the next sections I will try to give an answer as to what are the reasons for these, honestly spoken, moderate results. Concretely, I will argue that this is mainly due to noise, present in differing intensities and at various levels in the data I acquired, that the model has to cope with:

register noise The textual styles vary greatly from iutilizing complex, hypotactic sentence structures (e.g. XQuAD), to highly informal, elliptic — even erratic — structures (e.g. SCARE).
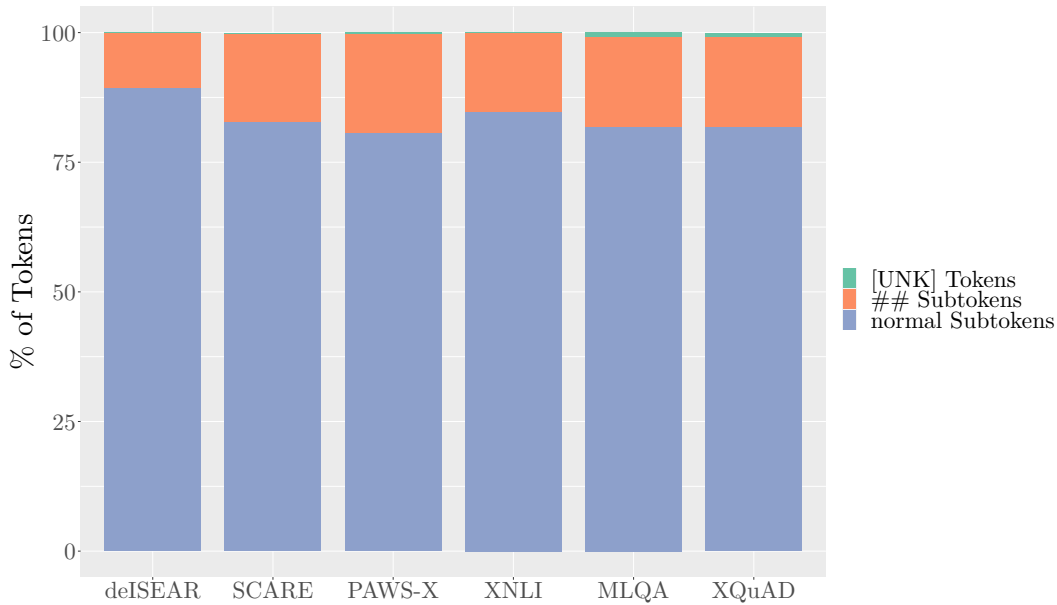
Figure 22: Percentages of token-types in all data sets. ## Subtokens represent the amount of tokens that get re-merged in the merged settings (e.g. "Master" "##arbeit" → "Masterarbeit". The amount of tokens that lie outside of the german BERT vocabulary is in all data sets extremely small (for deISEAR and XNLI there are no [UNK]s at all); the largest shares of such tokens are present in MLQA and XQuAD with .79% and .73%, respectively.

label noise Many of my data sets were constructed either automatically (e.g. scrambling text automatically to create paraphrase pairs) or employing crowd-sourcing techniques. Either way, the process is prone to erros. There are, e.g., 84 sentence pairs in the trainig set of PAWS-X that are 100% identical, yet labelled as non-paraphrases.

translation noise Due to the mostly employed semi-automatic translation approach for creating the various data sets, errors have been introduced into the data ranging from typical translation errors (e.g. english "bishop" in the clerical context translated to the german chess figure counterpart "Läufer", not "Bischof") to eventually wrongly copied labels, since the overall meaning changed during the tranlsation process (e.g. a sentence pair is no more contradictive but neutral).

SRL noise The SRLs obtained from DAMESRL are, conservatively formulated, questionable in their quality (e.g. modifiers are completely missing).

In short — the old GIGO concept from informatics holds mutatis mutandis also in NLP.

5.2 Register Noise

5.3 Label Noise

As [**?**] point out,

PAWS-X sentence number 45061, labelled as non-paraphrases:

Riverton was a parliamentary electorate in the New Zealand region of Southland .
Riverton was a parliamentary electorate in the New Zealand region of Southland .

Riverton war ein Parlamentswähler in der neuseeländischen Region Southland. River-
ton war ein Parlamentswähler in der neuseeländischen Region Southland.

5.4 Translation Noise

XNLI labelled as entailment

and that's a lot of it is due to the fact that the mothers are on drugs The mothers
take drugs.

Und vieles davon liegt daran, dass die Mütter Medikamente nehmen. Die Mütter
nehmen Drogen.

PAWS-X; different repair-strategies → different labels (gold: false)

Sawyers autorisierte Biografie wurde 2014 von Huston Smith veröffentlicht. Im Jahr
2014 wurde Huston Smith eine autorisierte Biographie von Sawyer veröffentlicht.

Im Jahr 2014 wurde «Huston Smith», eine autorisierte Biographie von Sawyer,
veröffentlicht.

Im Jahr 2014 wurde von—für—durch—trotz—wegen Huston Smith eine autorisierte
Biographie von Sawyer veröffentlicht.

Im Jahr 2014 wurde Huston Smith eine autorisierte Biographie von Sawyer veröffentlicht.

5.5 SRL Noise

A major question arising in the context of using automatically assigned Semantic
Roles in downstream tasks, is how good these Semantic Roles are. Since there is

no gold standard available for Semantic Role Labels for the data sets I use in my experiments, there is no straight-forward way to evaluate their quality automatically. In contrast to other tagging tasks like POS prediction or NER, Semantic Roles are not as black and white: While it is relatively easy to decide if a predicted POS tag is correct or incorrect, it is more a scale concerning SRLs.
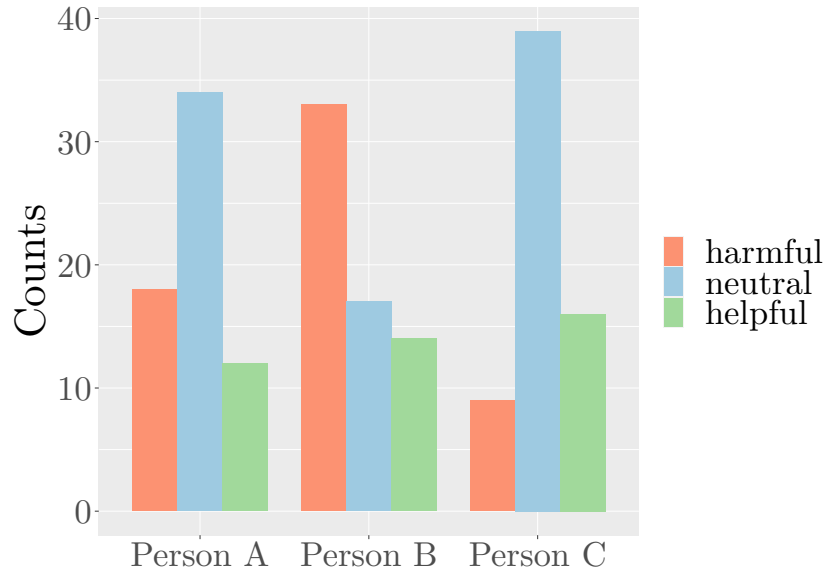
Figure 23: Independent evaluation of SRL quality by three people. Regardless of the label attributed to each example, it is obvious, that the total amount of sentences for which the annotators evaluated the corresponding semanti roles as *helpful*, is relatively stable.

Fleiss' $\kappa = 0.2048$ — this slightly above the threshold of «fair agreement», as defined by [Landis and Koch, 1977] (0.20).

The κ for helpful vs. other is even worse: 0.1944

for individual datasets:

deISEAR: 0.0814

SCARE: 0.2401

PAWS-X: 0.1245

XNLI: 0.2475

MLQA: -0.3636

XQuAD: -0.5

Do et al. [2018]

Figure 24: Estimated quality of SRLs per data set.

5.6 Ablation study

To be able to make substantial claims about the positive influence about a new algorithm over an established one, it is common ground to conduct an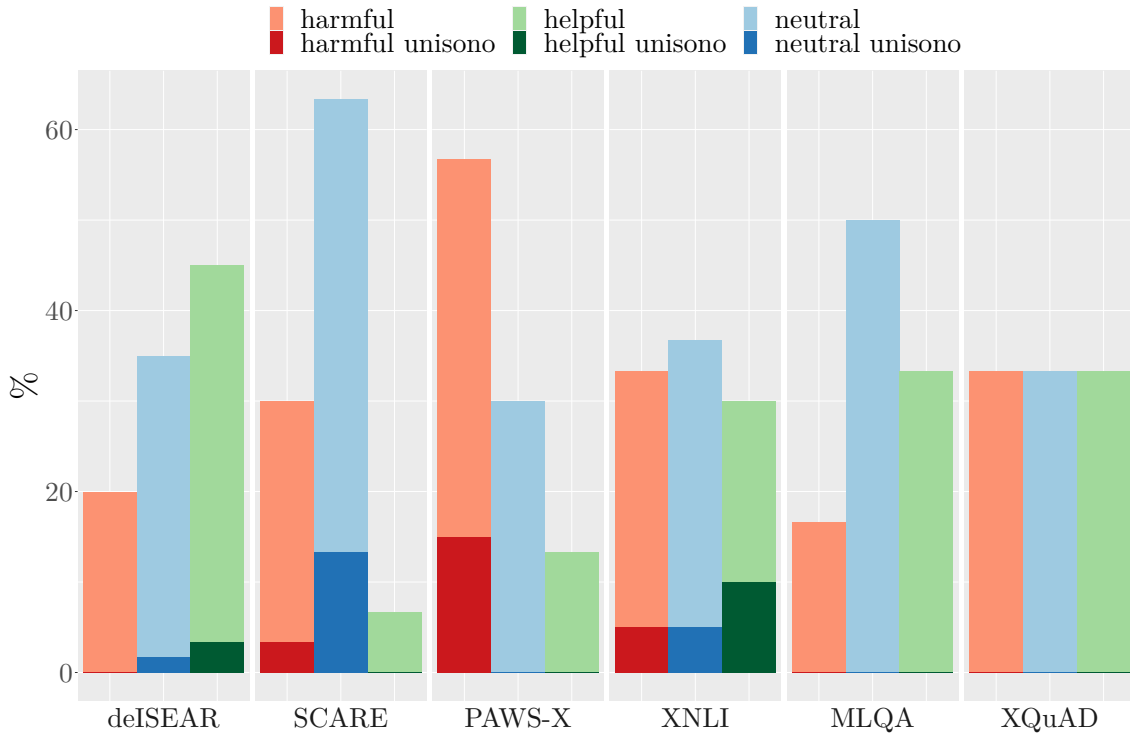 ablation study. In such a study, one tries to determine which aspects of the proposed architecture contribute how much to the overall performance gain (or loss, respectively).

In my case, i.e. the attempt to improve the performance of BERT regarding NLU tasks, the following question would need some ablation experiments to be answered: What part of the SRLs is most responsible for the performance boost? To be able to formulate this in a matter which can be experimentally tested, I identify two easily separatable and testable aspects of SRLs: Firstly, the information what parts of a sentence are the predicates. The intuition behind this is that maybe the head relies mostly on the information as to which tokens carry information about the events that happen in a given sentence. To test this, I simply drpo the information about all SRLs, except the information that a token is a predicate. In the second case, the hypothesis is reversed: Maybe the head is able to get the most useful hints about the information which inidcates what role certain token groups play in a given sentence.

To test for this all information about predicates is dropped and only information about arguments is preserved.

```
Ich        B-A0        O
weiß       B-V         O
nicht      O           O
ob         B-A1        O
er         I-A1        B-A0
danach     I-A1        O
in         I-A1        B-A1
Augusta    I-A1        I-A1
geblieben  I-A1        B-V
ist        I-A1        O
.          O           O
==============================
Er         B-A0
wohnte     B-V
weiterhin  O
in         B-A1
Augusta    I-A1
.          O
```

SRL 5.1: Normal SRLs.

```
Ich        O    O         Ich        B-A0      O
weiß       B-V  O         weiß       O         O
nicht      O    O         nicht      O         O
ob         O    O         ob         B-A1      O
er         O    O         er         I-A1      B-A0
danach     O    O         danach     I-A1      O
in         O    O         in         I-A1      B-A1
Augusta    O    O         Augusta    I-A1      I-A1
geblieben  O    B-V       geblieben  I-A1      O
ist        O    O         ist        I-A1      O
.          O    O         .          O         O
============================      ==============================
Er         O              Er         B-A0
wohnte     B-V            wohnte     O
weiterhin  O              weiterhin  O
in         O              in         B-A1
Augusta    O              Augusta    I-A1
.          O              .          O
```

SRL 5.2: Left: Only predicate SRLs. **Right**: Only argument SRLs.

		−SRL	+SRL		
			only PREDs	only ARGs	normal
deISEAR α	FFNN Head subtok. zeros	*70.86*	72.19	<u>75.50**</u>	**77.48**
SCARE α	[CLS] Head merged duplicate	*83.33*	84.47	<u>85.23</u>	**85.61***
PAWS-X β	[CLS] Head merged duplicate	*79.92*	80.53	<u>80.68</u>	**82.51***
XNLI β	GRU Head subtok. zeros	*66.84*	67.02	**68.00**	<u>67.82</u>

Table 16: Ablation on Effect of PREDs and ARGs isolated. note that PRED/ARG SRL not significant (SCARE, XNLI ARGs almost, ca. 11%)

6 Conclusion

In this project we have done so much.[1]
We could show that . . .

6.1 Outlook / Future Work

- better SRL system

- more thoroughful SRL quality estimation

---

[1]Thanks to many people that helped me.

Glossary

Of course there are plenty of glossaries out there! One (not too serious) example is the online MT glossary of Kevin Knight [2] in which MT itself is defined as

> techniques for allowing construction workers and architects from all over the world to communicate better with each other so they can get back to work on that really tall tower.

accuracy A basic score for evaluating automatic **annotation tools** such as **parsers** or **part-of-speech taggers**. It is equal to the number of **tokens** correctly tagged, divided by the total number of tokens. [. . .]. (See **precision and recall**.)

clitic A morpheme that has the syntactic characteristics of a word, but is phonologically and lexically bound to another word, for example *n't* in the word *hasn't*. Possessive forms can also be clitics, e.g. The dog*'s* dinner. When **part-of-speech tagging** is carried out on a corpus, clitics are often separated from the word they are joined to.

---

[2]Machine Translation Glossary (Kevin Knight): `http://www.isi.edu/natural-language/people/dvl.html`

References

N. Aepli. *Parsing Approaches for Swiss German*. PhD thesis, University of Zurich, 2018.

M. Artetxe, S. Ruder, and D. Yogatama. On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856*, 2019.

S. Buchholz and E. Marsi. Conll-x shared task on multilingual dependency parsing. In *Proceedings of the tenth conference on computational natural language learning (CoNLL-X)*, pages 149–164, 2006.

K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

A. Conneau, G. Lample, R. Rinott, A. Williams, S. R. Bowman, H. Schwenk, and V. Stoyanov. Xnli: Evaluating cross-lingual sentence representations. *arXiv preprint arXiv:1809.05053*, 2018.

M.-C. De Marneffe, T. Dozat, N. Silveira, K. Haverinen, F. Ginter, J. Nivre, and C. D. Manning. Universal stanford dependencies: A cross-linguistic typology. In *LREC*, volume 14, pages 4585–4592, 2014.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Q. N. T. Do, A. Leeuwenberg, G. Heyman, and M. F. Moens. A flexible and easy-to-use semantic role labeling framework for different languages. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 161–165, 2018.

R. Dror, G. Baumer, S. Shlomov, and R. Reichart. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, 2018.

K. A. Foth. Eine umfassende constraint-dependenz-grammatik des deutschen. 2006.

K. Gerdes and S. Kahane. Word order in german: A formal dependency grammar using a topological hierarchy. In *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pages 220–227, 2001.

D. Gildea and D. Jurafsky. Automatic labeling of semantic roles. *Computational linguistics*, 28(3):245–288, 2002.

A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009, 2009.

T. Groß and T. Osborne. The dependency status of function words: Auxiliaries. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, pages 111–120, 2015.

J. Hajič, M. Ciaramita, R. Johansson, D. Kawahara, M. A. Martí, L. Màrquez, A. Meyers, J. Nivre, S. Padó, J. Štepánek, et al. The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. 2009.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8): 2554–2558, 1982.

D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits. Is bert really robust? natural language attack on text classification and entailment. *arXiv preprint arXiv:1907.11932*, 2, 2019.

M. Johnson. How the statistical revolution changes (computational) linguistics. In *Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, pages 3–11, 2009.

D. Jurafsky and J. H. Martin. Speech and language processing (draft). october 2019. *URL https://web. stanford. edu/˜ jurafsky/slp3*, 2019.

P. Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 388–395, 2004.

M. Kracht. Introduction to linguistics. *Departement of*, 2007.

J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

P. Lewis, B. Oğuz, R. Rinott, S. Riedel, and H. Schwenk. Mlqa: Evaluating cross-lingual extractive question answering. *arXiv preprint arXiv:1910.07475*, 2019.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

W. Morgan. Statistical hypothesis tests for nlp, 2005.

B. Myagmar, J. Li, and S. Kimura. Transferable high-level representations of bert for cross-domain sentiment classification. In *Proceedings on the International Conference on Artificial Intelligence (ICAI)*, pages 135–141. The Steering Committee of The World Congress in Computer Science, Computer . . . , 2019.

M. Palmer, D. Gildea, and N. Xue. Semantic role labeling. *Synthesis Lectures on Human Language Technologies*, 3(1):1–103, 2010.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

A. Rogers, O. Kovaleva, and A. Rumshisky. A primer in bertology: What we know about how bert works, 2020.

T. Samardzic. *Dynamics, causation, duration in the predicate-argument structure of verbs: a computational approach based on parallel corpora.* PhD thesis, University of Geneva, 2013.

M. Sänger, U. Leser, S. Kemmerer, P. Adolphs, and R. Klinger. Scare—the sentiment corpus of app reviews with fine-grained annotations in german. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1114–1121, 2016.

K. R. Scherer and H. G. Wallbott. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2):310, 1994.

A. Schiller, S. Teufel, C. Stöckert, and C. Thielen. Guidelines für das tagging deutscher textcorpora. *University of Stuttgart/University of Tübingen*, 1999.

G. Schneider. *Hybrid long-distance functional dependency parsing.* PhD thesis, University of Zurich, 2008.

R. Sennrich, M. Volk, and G. Schneider. Exploiting synergies between open resources for german dependency parsing, pos-tagging, and morphological analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 601–609, 2013.

R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

E. Troiano, S. Padó, and R. Klinger. Crowdsourcing and validating event-focused emotion corpora for german and english. *arXiv preprint arXiv:1905.13618*, 2019.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

A. Warstadt, A. Singh, and S. R. Bowman. Neural network acceptability judgments, 2019.

A. Williams, N. Nangia, and S. R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*, 2017.

E. Wittenberg. *With light verb constructions from syntax to concepts*, volume 7. Universitätsverlag Potsdam, 2016.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. Transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

Y. Yang, Y. Zhang, C. Tar, and J. Baldridge. Paws-x: A cross-lingual adversarial dataset for paraphrase identification. *arXiv preprint arXiv:1908.11828*, 2019.

A. Yeh. More accurate tests for the statistical significance of result differences. *arXiv preprint cs/0008005*, 2000.

Y. Zhang, J. Baldridge, and L. He. Paws: Paraphrase adversaries from word scrambling. *arXiv preprint arXiv:1904.01130*, 2019a.

Z. Zhang, Y. Wu, H. Zhao, Z. Li, S. Zhang, X. Zhou, and X. Zhou. Semantics-aware bert for language understanding. *arXiv preprint arXiv:1909.02209*, 2019b.

Lebenslauf

Persönliche Angaben

Jonathan Schaber

Schartenstrasse 103

5430 Wettingen

jonathan.schaber@uzh.ch

Schulbildung

2006-2009	Fachmittelschule (FMS) Kantonsschule Wettingen
2009-2011	Matura Kantonsschule Wettingen
2012-2016	Bachelor-Studium Germanistik, Philosophie an der Universität Zürich
seit 2017	Master-Studium Computerlinguistik, historische Linguistik an der Universität Zürich

Berufliche und nebenberufliche Tätigkeiten

2012–2013	Tutorate PCL I+II

A Tables

Part of speech	POS type	POS	number of labels in my corpus
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	DET	**35**	280
14	Total	**35**	280

Table 17: Some very large table in the appendix

B List of something

This appendix contains a list of things I used for my work.

- apples
  - `export2someformat`
- bananas
- oranges
  - `bleu4orange`
  - `rouge2orange`