Originally published in: Preliminary proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019), October 9 – 11, 2019 at Friedrich-Alexander-Universität Erlangen-Nürnberg. - München [u.a.]: German Society for Computational Linguistics & Language Technology und Friedrich-Alexander-Universität Erlangen-Nürnberg, 2019. Pp. 130-139

# Detecting the boundaries of sentence-like units on spoken German

**Josef Ruppenhofer**
Institut für Deutsche Sprache
R5, 6-13
D-68161 Mannheim
ruppenhofer@ids-mannheim.de

**Ines Rehbein**
Institut für Deutsche Sprache
R5, 6-13
D-68161 Mannheim
rehbein@ids-mannheim.de

## Abstract

Automatic division of spoken language transcripts into sentence-like units is a challenging problem, caused by disfluencies, ungrammatical structures and the lack of punctuation. We present experiments on dividing up German spoken dialogues where we investigate the impact of task setup and data representation, encoding of context information as well as different model architectures for this task.

## 1 Introduction

Being able to structure natural spoken discourse into sentence-like units (SLUs) is desirable not only from a theoretical point of view, but is also a key requirement for enabling research in corpus linguistics as well as the application of Natural Language Processing tools (e.g. POS-tagging and parsing) to transcripts of spoken language. While various proposals have been made for how to divide spoken language in corpora into smaller units, typically these divions were not guided by syntactic considerations. Instead, division into inter-pausal units is common (e.g. Hamaker et al. (1998) for the Switchboard corpus). Until recently, for most languages no well-established system existed for detecting boundaries between sentence-like units that is both theoretically well-founded and practically operationalizable for large and diverse corpora of spoken interaction.

For German, the SegCor project (Westpfahl and Gorisch, 2018; Westpfahl et al., 2019) developed guidelines for dividing transcibed speech into sentence-like units using the topological field model of German surface syntax. Schmidt and Westpfahl (2018) subsequently presented a corpus-based study on how well the length of gaps between utterances can predict the syntactic boundaries annotated in the SegCor corpus.

In this work, we take up the challenge of automatically detecting boundaries between SLUs on the spoken German of the SegCor transcripts. Further, we apply our system not only to the question whether a gap, a long silence, coincides with a syntactic boundary but to all boundaries in general, including the ones that occur in continuous speech, such as interruptions and aborted utterances.

This paper proceeds as follows. We discuss related work in section 2 and present our dataset in sesction 3. In sections 4 and 5 we discuss the task formulations we employ and the features we use. Our experiments and their results are described in section 6, followed by a conclusion in section 7.

## 2 Related Work

In the realm of medially written language, the most closely related task is sentence boundary detection (SBD). Typically, this has been framed as deciding for a closed class of interpunctuation symbols (mainly '.','?','!') whether they represent the end of a sentence or not, with abbreviations constituting one of the key sources of error. While traditionally very high accuracies were reported, Read et al. (2012) show in their overview of SBD that performance can be significantly worse on text other than news, with machine learning-based systems often being less robust than rule-based or hybrid sytems. Comparing Wikipedia pages to topically related blogs, they also show that within the same domain, sentence-boundary detection performs less well the more informal the text type is. Read et al. (2012) observe that the traditional framing of the problem overlooks all the cases where sentences or rather sentence-like units, text sentences in the sense of Nunberg (1990), end without a punctuation symbol: on the 'standard' texts in their collection, this affects 12.3% of sentences. Read et al. (2012) therefore argue for a more general approach 'which considers the positions after every character as a potential boundary point'.

In the domain of medially spoken language, the detection of sentence-like units may use both textual and prosodic features. Gotoh and Renals (2000) performed experiments with HMMs on reference transcripts from BBC radio and tv programs which included repeated and incorrect speech as well as disfluencies. They also constructed an alternative pause duration model alone based on speech recogniser output aligned with the transcripts. The pause duration model outperformed the language modelling approach, while a combination of the two models provided further performance gains. Precision and recall scores of over 70% were attained for the task of deciding for each word whether it represents the last word of a sentence. In his work on sentence boundary detection on Czech radio news and discussion programs, Kolář (2008) similarly finds that combining several models works best.

Liu et al. (2005) evaluate the performance of a CRF-model on two English corpora (conversational telephone speech and broadcast news speech) on both human transcriptions and automatic speech recognition output. Their experiments show that the use of prosody improves performance over the use of word n-grams alone and that the addition of further features e.g. on pos-tags provides another improvement.

Roark et al. (2006) use a re-ranking approach to the detection of SLU boundaries. In a two stage approach, they first fix a subset of the word boundaries as points of division, yielding subsequences betwen fixed points, which they call fields. In the second stage, candidate boundaries within the fields are generated and then ranked.

In our own experiments, we will experiment with various features and task paramaters used by prior work such as e.g. POS, gap/pause-length, use of left and/or right context etc. In addition, we also explore extra features available with our dataset.

## 3 Dataset

The data used here is unlike most of the material used in related work in that it represents conversational speech that was furthermore recorded in non-laboratory settings. Also, it is characterized by interactions between two or more speakers. Since tools based on the automatic processing of the audio signal do not work all that well on our data, we instead work with the transcripts only. Our dataset consists of 33 documents with more than 54,000 lexical tokens originating from the FOLK corpus (Schmidt, 2014) that were divided into sentence-like units by the SegCor project. This data set was doubly annotated and disagreements were adjudicated (Westpfahl and Gorisch, 2018). Note that to avoid confusion, we reserve the term ”segment” and related forms for the division of speech into chunks by the transcribers that was guided by silences in the speech signal. For the division of the material into sentence-like units we will use the term “SLU boundary detection”.

The raw FOLK transcripts, which we take as our input and which lack SLU-boundaries, follow the cGAT conventions (Schmidt et al., 2015). Accordingly, the data uses ”contributions” and ”segments” as the fundamental units in the data structure. Segments of speech are the original units of transcription: transcribers are instructed to select them as chunks that can be transcribed in one go given cognitive load and useability of the transcription environment. Crucially, segment boundaries should be placed at word boundaries or at the beginning or end of pauses. Like segments, contributions are defined without any reference to syntactic considerations (Schmidt et al., 2015, 8):

> ‘A contribution in a cGAT transcript comprises all immediately consecutive segments attributed to a speaker. Contributions should not be confused with sentences, which are units of written language. Instead, they are to be understood as dialogue contributions.

`Pauses` (silences up to 0.2s) may occur between separate contributions but also within a contribution. `Gaps`, silences longer than 0.2s, always separate contributions in cGAT.

The relation between the input representation in terms of contributions and the intended output representation in terms of sentence-like units is not always one to one. Common deviations are as follows. First, a contribution may correspond to several SLUs as illustrated by (1).

(1) 1 contribution : *n* SLUs

a. $< c >$h ich weiß net ich glaub eher nich h h$< /c >$

b. $< s >$h ich weiß net$< /s >$
$< s >$ ich glaub eher nich h h$< /s >$

c. ‘I don’t know. I rather think not.’

Second, several contributions may jointly correspond to one SLU.

(2) *n* contributions : 1 SLU

a. $< c >$der beschäftigt sich$< /c >$
$< c >$(0.85) $< /c >$
$< c >$zwei minuten mit dem$< /c >$

b. $< s >$ der beschäftigt sich (0.85) zwei minuten mit dem $< /s >$

c. 'He occupies himself with that one for two minutes.'

Both situations may also occur in combination so that we get $n : m$-relations between contributions and SLUs.

To decide on SLU boundaries, we can use not only the transcribed word forms but also some further kinds of information about the tokens, which we will use as features (cf. section 5). Further, while we do not use acoustic features such as word durations and pitch contours, the transcript does give us access to temporal information that has proved useful in previous work (Gotoh and Renals, 2000). We encode pause length and, since we know which tokens are produced by which speaker, we also introduce turn boundaries into our representation.

## 4 Task formulations

We can approach the SLU boundary detection problem in various different ways. We discuss the major points of variation in what follows.

### 4.1 Granularity

In one line of experiments (coarse), we predict only whether a token is followed by some type of syntactic boundary (B) or not (O). In another line (fine), we also distinguish between several types of boundaries. From Westpfahl and Gorisch (2018), we adopt the following B(oundary) types.

**S** Simple sentential units consist of exactly one clause. In terms of word order, the clause may be of any of the types V1 (verb initial), V2 (verb second), V1/2 (cases that are unclear between V1 and V2) or in rare cases VL (verb last). The clauses may not have any dependent clauses.

**C** Complex sentential units consist of several clauses that are dependent on one another: Main clauses with subordinate clauses or relative clauses, conditional sentences, reported speech, and matrix-clause with sentient-verbs, complex pre-pre-fields with main clause, discontinuous sentences, and coordinated sentences if and only if the second sentence shows subject or verb ellipsis.

**N** Non-sentential units are all units that are not structured by a finite verb.

**A** An utterance which is disrupted, i.e. it opens a projection that subsequently goes unfilled.

**U** Tokens at the end of a unit whose status could not be categorized as one of the previous four cases.

Since in the context of sequence labeling we need to have a label on every token, we add several further categories of non-boundary labels. In the binary setting, these categories are merged into the non-boundary class (O).

**O** Words spoken by one of the speakers that are not followed by a boundary.

**X** is used for different types of non-verbal information: a) speaker turns, and b) pauses. We distinguish between pauses shorter than 0.2 sec and longer pauses. According to cGat, longer pauses always occur between two adjacent contributions and are not assigned to any speaker while shorter pauses are considered to be part of one speaker's contribution. For instance, the pause in (i) is part of speaker RD's contribution as they are just pausing speech for the purposes of word finding. By contrast, the pause in (ii) is not assigned to either speaker: it is clear that speaker RD has finished their turn, but speaker LH has not yet taken the floor.

i RD: ich könnte es ja darüber lösen dass ich das nicht auf das <pause> ko auf die konten der seefahrer buch sondern auf ein verrechnungskonto
'Well, I could fix it in this way that I don't book it on the acc on the accounts of the sailor but instead to a clearing account'

ii RD: ich versthe nichts davon
'I don't know anything about it'
<pause>
LH: okay. ...

In our experiments, both pause types are assigned the tag "X".

### 4.2 Views

Since our data comes from multi-party conversation it lends itself to two views. On the one hand, we can think of it as an integrated **conversation**, where contributions of speakers alternate, with occasional overlaps. The intuition behind adopting this view on the data is that a speaker's productions do depend on / respond to what the other speaker says. For instance, responses to questions are often not complete sentential units whether simple or complex but rather consist of non-sentential material. For that reason, it seems important to take into account what interlocutors are saying.

A second, complementary view of the data treats it as a set of **tracks** of speech, each by one specific person. The intuition behind this view is that the sentence-like units are local only to the given speaker's utterances. For instance, whether a sentence is simple or complex depends only on what the current speaker produces. In adopting a track view (track), we completely ignore the other speaker's productions.

Both views potentially have problems handling certain kinds of so-called split utterances (Purver et al., 2009). On the conversation view, utterances that are distributed across multiple contributions of the same speaker may be interrupted by contributions of other speakers. On the track view, utterances that are distributed across speakers (that is, co-constructed turns begun by one speaker but finished by another) cannot be recovered.

### 4.3 Instance creation

We define instances for the classifier either in terms of word **windows** of varying size or in terms of N **merged** contributions.[1]

### 4.4 Model type

As demonstrated by the related work, one established way to approach the SLU boundary detection problem is in terms of **sequence labeling**. The task consists in algorithmically assigning a categorical label to each item in a sequence of observed values. In our task, a token is labeled either as being followed by a boundary or not.

As a baseline approach, we adopt a classical Conditional Random Fields (Lafferty et al., 2001) tagger, using the CRFsuite implementation by Okazaki (2007), for which we provide our own feature engineering.

We compare this system with two more recent neural architectures. The first system is an implementation of the model of Lample et al. (2016), using biLSTMs for input encoding, based on word and character-based embeddings, followed by a CRF layer on top (Reimers and Gurevych, 2017).[2] The second model, the flair sequence tagger (Akbik et al., 2019), has a similar architecture that also combines biLSTMs and a CRF layer on top. In addition, flair uses *contextual string embeddings* (Akbik et al., 2018) which model words as contextualized sequences of characters, resulting in different embeddings for the same string, depending on its surrounding context.

## 5 Features

The data encodes the following information that we can use as features in our experiments.

**Tokens** The simplest feature are the raw transcribed tokens.

**POS** The SegCor data includes automatically predicted POS tags.

**Normalization** The normalization layer contains the canonicalized form for the raw tokens. For instance, when an instance of the first person present form of the verb *verstehen* 'understand' is pronounced as two syllables, without its final weak syllable, it is transcribed as *versteh*. The normalization of the token will be the expected canonical form *verstehe*. Also while all noun tokens appear lowercased in the transcription, they are written with initial capitals on the normalization layer.

**Lemma** The lemma forms for the transcribed data.

## 6 Experiments

At the highest level, we divide our experiments depending on the granularity, coarse or fine. Within these high-level groups, we discuss the experiments in sets that address a common research question.

We use 70, 10 and 20% of the data for training, development and testing, respectively. We do not split up individual transcriptions but put them whole into either train, dev or test. This makes the task slightly harder as we test on data from new speakers that have not been seen during training, and on new topics that are not included in

[1] Other variations are possible such as creating overlapping instances. For instance, with word windows we could create one instance from words 1-10 and the next from words 2-11 etc. We could proceed similarly in the case of contributions.

[2] `https://github.com/UKPLab/emnlp2017-bilstm-cnn-crf/`

| ID | View | Instances | Macro Acc | Macro F1 | F1 B | F1 O | Description |
|---|---|---|---|---|---|---|---|
| 1 | track | single | 83.75 | 45.58 | 0.00 | 91.16 | majority class, i.e. no boundaries |
| 2 | track | single | 89.98 | 74.99 | 55.63 | 94.35 | boundary at end of contribution |

Table 1: Results for rule-based baselines (coarse-grained, track: track-view; singe: single contributions)

|  | ID | View | Instances | Macro Acc | Macro F1 | F1 B | F1 O | context | features |
|---|---|---|---|---|---|---|---|---|---|
| instance creation | 3 | track | single | 94.20 | 87.25 | 77.84 | 96.67 | +/-2 | word,pos |
|  | 4 | track | single | 93.66 | 86.04 | 75.73 | 96.36 | +/-1 | word,pos |
|  | 5 | track | merged | 94.69 | 88.33 | 79.71 | 96.95 | +/-2 | word,pos |
|  | 6 | track | merged | 93.99 | 86.74 | 76.93 | 96.54 | +/-1 | word,pos |
|  | 7 | track | window | 94.01 | 86.58 | 76.59 | 96.57 | +/-2 | word,pos |
|  | 8 | conv. | window | 93.54 | 85.42 | 74.54 | 96.30 | +/-2 | word,pos |
| context size | 9 | track | merged | 94.78 | 88.56 | 80.13 | 97.00 | +2 | word,pos |
|  | 10 | track | merged | 93.53 | 85.60 | 74.90 | 96.29 | +1 | word,pos |
|  | 11 | track | merged | 89.21 | 73.25 | 52.58 | 93.91 | -1 | word,pos |
|  | 12 | track | merged | 88.75 | 72.86 | 52.09 | 93.63 | -2 | word,pos |
| single feats. | 13 | track | merged | 93.86 | 85.87 | 75.25 | 96.50 | +/-2 | word |
|  | 14 | track | merged | 93.86 | 86.46 | 76.46 | 96.47 | +/-2 | pos |
|  | 15 | track | merged | 93.76 | 85.89 | 75.36 | 96.43 | +/-2 | lemma |
|  | 16 | track | merged | 94.16 | 86.88 | 77.10 | 96.66 | +/-2 | normalization |
| norm. | 17 | track | single | 94.14 | 87.15 | 77.68 | 96.63 | +/-2 | norm, pos |
|  | 18 | track | merged | 94.78 | 88.52 | 80.05 | 97.00 | +/-2 | norm, pos |
| turn | 19 | track | merged | 92.56 | 84.38 | 73.07 | 95.68 | +/-2 | word, pos; no turns |

Table 2: Results for sequence labeling with CRFsuite (coarse-grained, track-view; conv.: conversation; merged: 5 merged contributions; window: 10-word windows)

| ID | View | Instances | Macro Acc | Macro F1 | F1 B | F1 O | Embeddings | Schema |
|---|---|---|---|---|---|---|---|---|
| 20 | track | merged | 94.14 | 87.06 | 77.48 | 96.63 | Reimers2017 | word |
| 21 | track | merged | 94.36 | 87.69 | 78.63 | 96.75 | Reimers2017 | norm |

Table 3: Results for biLSTM-CRF sequence tagger (Lample et al., 2016) (coarse-grained, track-view)

| ID | View | Instances | Macro Acc | Macro F1 | F1 B | F1 O | Embeddings |
|---|---|---|---|---|---|---|---|
| 22 | track | merged5 | 95.07 | 89.59 | 82.05 | 97.14 | fasttext+flair |
| 23 | track | merged5 | 92.28 | 83.42 | 71.30 | 95.54 | fasttext |
| 24 | track | merged5 | 94.83 | 89.28 | 81.56 | 97.00 | fasttext+custom |
| 25 | track | merged5 | 95.43 | 90.23 | **83.11** | 97.36 | fasttext+flair+custom |

Table 4: Results for flair's sequence tagger with contextual string embeddings (coarse-grained, track-view)

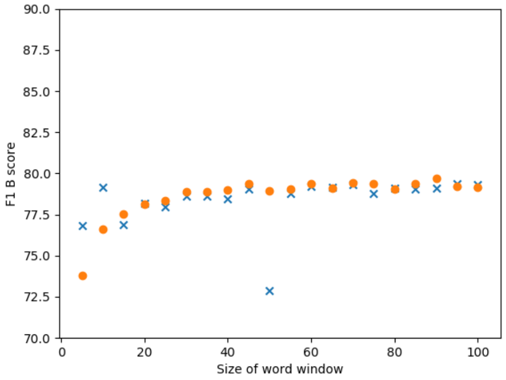

Figure 1: F1 B-score for word windows of various sizes (dots: conversations; x's: tracks; step size=5; CRFsuite)

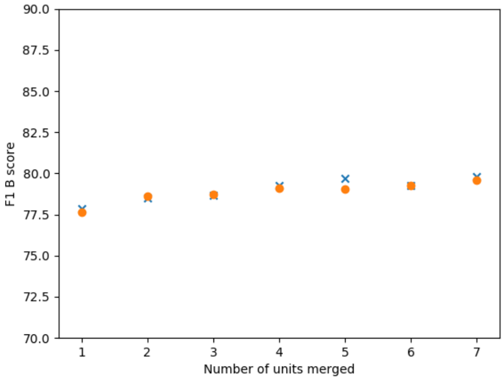

Figure 2: F1 B-score for track view in relation to contributions merged (CRFsuite) (dots: conversations; x's: tracks)

the training set. Thus, the classifier cannot adapt to speaker-specific features and might encounter a larger amount of unknown words. However, this setting is more realistic and will give us a better estimate of what to expect when applying our models to new data.

For all non-deterministic models, we report results averaged over three runs for each configuration.

### 6.1 Coarse-grained classification

**Baselines** In addition to using CRFsuite as a baseline, we calculated the following two rule-based baselines (table 1). Baseline 1 always assigns the majority class (no boundary) while baseline 2 predicts a boundary at the last token in each contribution. Recall that the contributions are not *gold* sentences but can also cross syntactic boundaries, which is shown by the less-than-perfect results for baseline 2 (89.98% acc. and 55.63% F1 for the **B**oundary class). As will be shown by the experiments to follow, machine-learning based systems, unsurprisingly, can yield much better results.

**Views and instance creation** First, we investigate the impact of view and instance creation on the performance for varying window sizes. Figure 1 plots the F1 scores for **B**oundaries relative to growing sizes of word windows used to construct instances. The results are very similar regardless of whether we use the conversational view or the track view.

Figure 2 shows the development of the F1 B-score in relation to the number of contributions that are assembled into one instance. We observe that, here too, the results hardly differ between the track view and the conversational view.

While it should not matter much in practice, we choose to mainly work with the combination of merging segments on the track view for the remainder of the paper since the highest F1-score that we obtained in these experiments come from this combination.

**Importance of Context** We now focus on the question where in the context the relevant information for boundary detection is. Thus, the second block of experiments varies the context relative to our reference experiments 5 and 6 (Table 2), using either only the left or the right context, or no context at all. The contrast between the results for the experiments with one-sided context shows that the right context is clearly more important than the left one and that the left context by itself does not hold very much information to begin with.

**Individual features** Experiments 13–16 present results for runs with individual features. The results show that not all forms of generalizing over the concrete tokens work equally well. The automatically assigned lemmatization probably is worst because on our data it is also often wrong. POS-tags are better but the normalized text representation, though also automatically assigned, is best.

**Normalization** Following on the observation about the utility of normalization, in experiments 17 and 18, we use the normalization layer instead of the transcribed tokens in combination with POS tags. When contrasting the results of these experiments with those of exp. 3 and 5, we see that normalization gives slightly better results only in the second setting. Given that normalization is also time-consuming, in later experiments we will not use the normalization layer but instead use the transcribed speech as input.

**Importance of Sequencing Information** In experiment 19, we use a version of the data from which, unlike for all other track view-based experiments, the representation of turns has been eliminated. Compared to the matched basic experiment 5, we see a significant drop in Macro F1 and the F1 for the B(oundary) class, which underscores the importance of including information on turns.

**Classical CRF vs. biLSTM-CRF** Recent advances in NLP have shown the expressive power of neural networks. We thus compare the performance of the classical CRF sequence tagger to two neural systems, the one of (Lample et al., 2016; Reimers and Gurevych, 2017) and the flair sequence tagger, as described in Section 4.4.

Table 3 shows that the neural biLSTM-CRF does not always improve results over the classical CRF. The first system uses word and charcter-based embeddings as features and predicts the binary labels $\{B, O\}$. This configuration does not outperform CRFsuite configurations such as 5 where we also use POS tags as features, in addition to the word tokens.

The biLSTM-CRF can make better use of the normalization, as shown in experiment 21. Compared to experiment 16, we gain 1.5% in performance. Both systems, however, are outperformed by the flair sequence tagger with contextual string embeddings (Table 4, exp 22).

**Embeddings used** Given that flair outperforms the model of Lample et al. (2016) despite their similar architecture, we now explore variation around the embeddings used in flair. Experiment 23 shows the value of flair's contextual string embeddings: without them performance decreases by more than 10% for F1 B (see exp. 22).

In our next experiment, we want to test whether we can increase performance by training our own contextual string embeddings on text that is more similar to our data. For this, we train flair embeddings for 20 epochs on ca. 11 million 'sentences' extracted from the open subtitles corpus (Lison and Tiedemann, 2016) and an in-house twitter dataset. These sentences were filtered to be at most 60 characters long and to contain no more than one comma and one period, question mark or exclamation mark. The punctuation marks were removed before training and the data was lowercased. In experiment 24 we use these custom embeddings in combination with fasttext only without the default forward and backward embeddings provided by the flair library.

The results show that the custom embeddings are quite good on their own (exp. 24). Combining them with flair's pretrained embeddings further improves results, showing that our custom embeddings contain complementary information (exp. 25). While the results suggest that the use of more domain-similar contextual string embeddings is beneficial, we cannot be sure that the improvements are really due to domain similarity. To test this in future work, we will need to compare our results to another type of custom embeddings trained on a corpus of equal size but with different properties that are less similar to spoken language, such as newspaper text.

## 6.2 Fine-grained classification

We now turn to the fine-grained setting which distinguishes between five kinds of boundary labels. For ease of presentation and since the non-boundary labels are not important to us, we will report F1 scores for each boundary label with the exception of the U(ninterpretable) class, which is conceptually ill-defined since by definition it is unclear whether, and what kind of, a boundary occurs. As well as the global Macro F1 and Macro Accuracy scores, we also report a score "Macro F1 B" which constitutes the macro average over the boundary labels, including U.

As a reference for the flair sequence tagger, Table 5 shows results for CRFsuite for the trackwise view and instances formed by merging contributions.[3] As shown by the difference in F1-scores between the fine-grained and the coarse-grained settings from Table 2, the fine-grained task is much harder. Again, using word windows of size 10 for instance creation is worse than merging contributions.

The gap between CRFSuite and the neural system shows the potential of the contextual string embeddings: Flair outperforms CRFSuite susbtantially (cf. exp. 29 vs. 27). Focusing on the flair results, we see that the performance on the individual boundary types strongly depends on their fre-

[3] For lack of space we do not report results for the biLSTM-CRF model of (Lample et al., 2016; Reimers and Gurevych, 2017) which again was outperformed by flair.

| Id | View | Instances | Macro F1 | Macro Acc | F1 A | F1 C | F1 N | F1 S | Macro F1 B |
|---|---|---|---|---|---|---|---|---|---|
| 26 | track | window | 58.51 | 97.61 | 22.79 | 26.32 | 73.55 | 51.01 | 43.42 |
| 27 | track | merged | 58.15 | 97.65 | 25.30 | 26.24 | 73.92 | 52.20 | 44.20 |

Table 5: Results for fine-grained sequence labeling with CRFsuite

| Id | View | Instances | Macro F1 | Macro Acc | F1 A | F1 C | F1 N | F1 S | Macro F1 B |
|---|---|---|---|---|---|---|---|---|---|
| 28 | track | window | 68.59 | 98.10 | 42.82 | 45.76 | 80.16 | 66.34 | 56.69 |
| 29 | track | merged5 | 70.24 | 98.22 | 42.93 | 50.49 | 81.59 | 68.95 | 58.98 |

Table 6: Results for fine-grained sequence labeling with flair

quency: results for the rarer classes A(borted) and C(omplex) are substantially lower than the ones for the more frequent classes N(on-sentential) and S(imple).

### 6.3 Error analysis

To get a sense of what the flair sequence tagger is able to learn, in Table 7 we take a look at the confusion matrix for the best fine-grained experiment 29. Among the boundary classes, A(borted) segments are mostly not recognized as having any kind of boundary, i.e. they receive the label O; smaller subsets of true A's are mistaken for non-sentential units or simple sentences. When A's get confused for O's, this often seems to be due to the boundary token being an incomplete, partial word such as *a* or *we*.

For C(omplex) segments, being mistaken for a simple sentence (S) is the most common error, before not being recognized as any kind of bounded segment. One class of C-S confusions arises when subordinate complement clauses lack a complementizer and verb-second word order is used, as in example (3).

(3) < *c* > ich wiederhole das sind tonsteine (.) mit eingelagerten kalksandsteinbänke < /*c* >
'I repeat [that] these are mudstones with embedded banks of sand-lime brick.'

|  | A | C | N | O | S | U | X | Total |
|---|---|---|---|---|---|---|---|---|
| A | 57 | 3 | 12 | 93 | 20 | 0 | 0 | 185 |
| C | 0 | 98 | 5 | 61 | 75 | 1 | 0 | 150 |
| N | 5 | 6 | 584 | 78 | 36 | 0 | 0 | 709 |
| O | 12 | 26 | 102 | 8836 | 84 | 2 | 0 | 9062 |
| S | 7 | 24 | 17 | 128 | 439 | 0 | 0 | 615 |
| U | 1 | 0 | 4 | 6 | 2 | 9 | 0 | 22 |
| X | 0 | 0 | 0 | 0 | 0 | 0 | 2105 | 2105 |

Table 7: Confusion matrix for best fine-grained run (exp. 29; across: predicted; down: gold)

Finally, for S(imple) sentences not being recognized as a bounded segment is the most common error. One subtype of this error that we recognize are cases where the final token is an unlikely one. Consider example 4, whose true labeling is given. The error that flair makes is to include all the tokens in a single S(imple) sentence, even though this means that the resulting simple sentence incorrectly has two finite verbs. Potentially, the error occurs because the adverb *angeblich* 'supposedly' is an unlikely sentence ending token. In example 5, the initial complex sentence is correctly recognized but the following simple sentence receives no boundary label even though it is followed by a change of turn. Again, the problem seems to be that the subject pronoun *er* 'he' is an unlikely sentence-final token. Other instances concern elliptical cases where modal verbs occur sentence-finally without an infinitival complement (e.g. *die müssen* 'They must'). A second subtype of error consists of infrequent sentence types. Consider the example in 6. This is an unusual case because it is a free-standing subordinate clause, which gets treated as a simple sentence according to the SegCor guidelines. Flair marks no boundary here, which results in the main clause of the following complex sentence having two finite verbs.

(4) < *s* >da war des doch fast die älteschte mutter angeblich< /*s* >< *s* >mit siebenungsechzig hat se s kind gekriegt oder so< /*s* >
'She was almost the oldest mother there supposedly. She had the child at sixty-seven or thereabouts.'

(5) $< c >$ was ich gelesen hab (.) muss immer derjenige äh zu lebzeiten schon seine einverständnis abgegben$< c/ >$ $< s >$nur die nimmt er$< /s >$
'From what I have read that person always has to give their consent during their lifteime. Only those ones he accepts.'

(6) $< s >$ ob ich des hinkriech $< /s >$
'[I am wondering] if I can manage that.'

Finally, we want to note that sentence boundary labeling cannot be done perfectly by humans and that its diffculty is variable across text types. Westpfahl and Gorisch (2018) report an average kappa of 0.69 across 8 transcripts. Across the transcripts, the kappa value ranges from 0.53 for a conflictual interaction to 0.76 for a reading child. While Westpfahl and Gorisch (2018) give no breakdown of which confusions among boundary types are most frequent for their human annotators, they do show a further complication of the task: the different sentence types are distributed differently across different text types and their specific properties also vary by text type. For instance, in so-called expert talk, simple sentences are longer than in other texts. Taken together, these considerations underline the challenge in the task we tackle.

## 7 Conclusions and Future Work

We have investigated the problem of detecting SLUs in spoken German. We found that the choice of data representation for the classifier is important: small word windows perform worse than larger ones but the merging of contributions performs well in a robust way, no matter the size. Further, we found that the main challenge of the task is to recognize sentence beginnings: the right context is much more important than the left context. We also verified that using information on turns is important. Finally, we found that augmenting flair's embeddings with domain-similar custom embeddings further enhances performance.

Given the success of the contextual string embeddings, in future work we would like to investigate whether other contextualized representations such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) can yield further improvements.

Another approach to SLU boundary detection frames it in terms of **sequence-to-sequence** learning, using attention-based neural encoder-decoder models (Bahdanau et al., 2015). Here, a model is trained to convert sequences from one domain to sequences in another domain. A typical application scenario for this class of models is machine translation. In our case, we would translate spoken German utterances lacking SLU boundaries into speech with SLU boundaries. While initial experiments showed that sequence-to-sequence models are also able to learn boundaries for spoken utterances, we did not have enough training data to achieve competetive results. We will pursue this avenue in future work, using additional naturalistic as well as synthetically created training data.

## Acknowledgments

This research has been partially supported by the Leibniz Science Campus "Empirical Linguistics and Computational Modeling", funded by the Leibniz Association under grant no. SAS-2015-IDS-LWC and by the Ministry of Science, Research, and Art (MWK) of the state of Baden-Württemberg.

## References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*, ICLR 2015.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *The 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT 2019, pages 4171–4186.

Yoshihiko Gotoh and Steve Renals. 2000. Sentence boundary detection in broadcast speech transcripts. In *in Proc. of ISCA Workshop: Automatic Speech Recognition: Challenges for the new Millennium ASR-2000*, pages 228–235.

Jonathan Hamaker, Yu Zeng, and Joseph Picone. 1998. Rules and guidelines for transcription and segmentation of the switchboard large vocabulary conversational speech recognition corpus. Technical report.

Jáchym Kolář. 2008. *Automatic Segmentation of Speech into Sentence-like Units*. Ph.D. thesis, PhD Thesis, University of West Bohemia, Pilsen, Czech Republic.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270. Association for Computational Linguistics.

Pierre Lison and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Yang Liu, Andreas Stolcke, Elizabeth Shriberg, and Mary Harper. 2005. Using conditional random fields for sentence boundary detection in speech. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 451–458. Association for Computational Linguistics.

Geoffrey Nunberg. 1990. *The Linguistics of Punctuation*. CSLI, 01.

Naoaki Okazaki. 2007. Crfsuite: a fast implementation of conditional random fields (crfs).

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT 2018, pages 2227–2237.

Matthew Purver, Christine Howes, Patrick G. T. Healey, and Eleni Gregoromichelaki. 2009. Split utterances in dialogue: A corpus study. In *Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, SIGDIAL '09, pages 262–271, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jonathon Read, Rebecca Dridan, Stephan Oepen, and Lars Jørgen Solberg. 2012. Sentence boundary detection: A long solved problem? In *Proceedings of COLING 2012: Posters*, pages 985–994, Mumbai, India, December. The COLING 2012 Organizing Committee.

Nils Reimers and Iryna Gurevych. 2017. Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 338–348, Copenhagen, Denmark, 09.

Brian Roark, Yang Liu, Mary Harper, Robin Stewart, Matthew Lease, Matthew Snover, Izhak Shafran, Bannie Dorr, John Hale, Anna Krasnyanskaya, and Lisa Yung. 2006. Reranking for sentence boundary detection in conversational speech. In *2006 IEEE International Conference on Acoustics, Speech, and Signal Processing - Proceedings*, volume 1, 12.

Thomas Schmidt and Swantje Westpfahl. 2018. A study on gaps and syntactic boundaries in spoken interaction. In Adrien Barbaresi, Hanno Biber, Friedrich Neubarth, and Rainer Osswald, editors, *The 14th Conference on Natural Language Processing*, KONVENS 2018, pages 40 – 49. Austrian academy of sciences, Vienna, Austria.

Thomas Schmidt, Wilfried Schütte, and Jenny Winterscheid. 2015. cgat. konventionen für das computergestützte transkribieren in anlehnung an das gesprächsanalytische transkriptionssystem 2 (gat2). Working paper, IDS Mannheim, Mannheim.

Thomas Schmidt. 2014. The research and teaching corpus of spoken german – folk. In *The 9th conference on international language resources and evaluation*, LREC 2014, pages 383 – 387, Reykjavik. European Language Resources Association (ELRA).

Swantje Westpfahl and Jan Gorisch. 2018. A syntax-based scheme for the annotation and segmentation of german spoken language interactions. In *The Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions*, LAW-MWE-CxG 2018, pages 109 – 120. Association for Computational Linguistics, Stroudsburg, PA, USA.

Swantje Westpfahl, Thomas Schmidt, Anton Borlinghaus, and Hanna Strub. 2019. Guideline: syntaktische segmentierung in folker. Working paper, Leibniz-Institut für Deutsche Sprache (IDS), Mannheim.