Er ... well, it matters, right? On the role of data representations in spoken language dependency parsing

Kaja Dobrovoljc
Jožef Stefan Institute
Ljubljana, Slovenia
kaja.dobrovoljc@ijs.si

Matej Martinc
Jožef Stefan Institute
Ljubljana, Slovenia
matej.martinc@ijs.si

Abstract

Despite the significant improvement of data-driven dependency parsing systems in recent years, they still achieve a considerably lower performance in parsing spoken language data in comparison to written data. On the example of Spoken Slovenian Treebank, the first spoken data treebank using the UD annotation scheme, we investigate which speech-specific phenomena undermine parsing performance, through a series of training data and treebank modification experiments using two distinct state-of-the-art parsing systems. Our results show that utterance segmentation is the most prominent cause of low parsing performance, both in parsing raw and pre-segmented transcriptions. In addition to shorter utterances, both parsers perform better on normalized transcriptions including basic markers of prosody and excluding disfluencies, discourse markers and fillers. On the other hand, the effects of written training data addition and speech-specific dependency representations largely depend on the parsing system selected.

1 Introduction

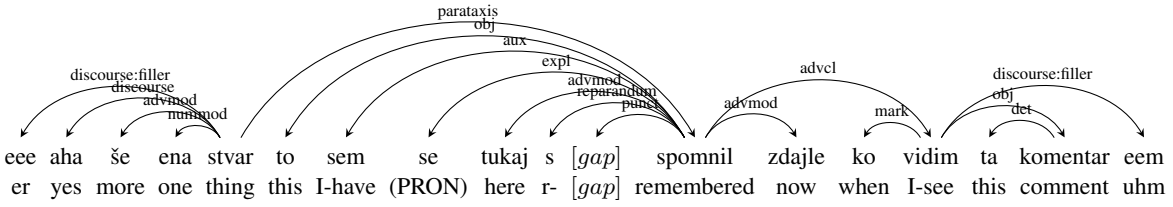
With an exponential growth of spoken language data available online on the one hand and the rapid development of systems and techniques for language understanding on the other, spoken language research is gaining increasing prominence. Many syntactically annotated spoken language corpora have been developed in the recent years to benefit the data-driven parsing systems for speech (Hinrichs et al., 2000; van der Wouden et al., 2002; Lacheret et al., 2014; Nivre et al., 2006), including two spoken language treebanks adopting the Universal Dependencies (UD) annotation scheme, aimed at cross-linguistically consistent dependency treebank annotation (Nivre, 2015).

However, in the recent CoNLL 2017 shared task on multilingual parsing from raw text to UD (Zeman et al., 2017), the results achieved on the Spoken Slovenian Treebank (Dobrovoljc and Nivre, 2016) - the only spoken treebank among the 81 participating treebanks - were substantially lower than on other treebanks. This includes the written Slovenian treebank (Dobrovoljc et al., 2017), with a best labeled attachment score difference of more than 30 percentage points between the two treebanks by all of the 33 participating systems.

Given this significant gap in parsing performance between the two modalities, spoken and written language, this paper aims to investigate which speech-specific phenomena influence the poor parsing performance for speech, and to what extent. Specifically, we focus on questions related to data representation in all aspects of the dependency parsing pipeline, by introducing different types of modifications to spoken language transcripts and speech-specific dependency annotations, as well as to the type of data used for spoken language modelling.

This paper is structured as follows. Section 2 addresses the related research on spoken language parsing and Section 3 presents the structure and annotation of the Spoken Slovenian Treebank on which all the experiments were conducted. Section 4 presents the parsing systems used in the experiments (4.1) and the series of SST data modifications to narrow the performance gap between written and spoken treebanks for these systems, involving the training data (4.3.1), speech transcriptions (4.3.2) and UD dependency annotations (4.3.3). Results are presented in Section 5, while conclusions and some directions for further work are addressed in Section 6.

Proceedings of the Second Workshop on Universal Dependencies (UDW 2018), pages 37–46
Brussels, Belgium, November 1, 2018.



(*oh yes one more thing I just r- [gap] remembered this here now that I see this comment*)

Figure 1: An example utterance taken from the Spoken Slovenian Treebank.

2 Related work

In line with divergent approaches to syntactic annotation of transcribed spoken data that either aim to capture the syntactic structure involving all uttered lexical phenomena in an utterance, or discard the (variously defined) noisy speech-specific structural particularities on the other, research into parsing spoken language can broadly be categorized in two main groups. On the one side of the spectrum, we find approaches that separate disfluences from parsing. Charniak and Johnson (2001) and Jørgensen (2007), for example, both report a significant increase in parsing the Switchboard section of the Penn Discourse Treebank (Godfrey et al., 1992), if disfluencies are first removed from the data. These two-pass pipeline approaches thus involve a separate task of automatic disfluency detection, one of the fundamental issues in automatic speech recognition (Liu et al., 2006; Lease et al., 2006).

Recently, however, several parsing systems using non-monotonic transition-based algorithms have emerged that enable joint parsing and disfluency detection (Honnibal et al., 2013; Honnibal and Johnson, 2015; Rasooli and Tetreault, 2013), showing that joint treatment of both problems can actually outperform state-of-the-art pipeline approaches (Honnibal and Johnson, 2014). These findings open a promising line of future research for the development of speech-specific parsing systems (Yoshikawa et al., 2016), especially those that also incorporate acoustic information (Kahn et al., 2005; Tran et al., 2017).

Nevertheless, apart from research on speech-specific parsing systems, very little research has been dedicated to other, data-related aspects of spoken language parsing. To our knowledge, with expection of Caines et al. (2017) and Nasr et al. (2014), who investigate the role of different types of training data used for parsing transcripts of speech, there have been no other systematic studies on the role of spoken data representations, such as transcription or annotation conventions, in spoken language parsing.

3 Spoken Slovenian Treebank

The Spoken Slovenian Treebank (Dobrovoljc and Nivre, 2016), which was first released as part of UD v1.3 (under the CC-BY-NC-SA 4.0 licence), is the first syntactically annotated collection of spontaneous speech in Slovenian. It is a sample of the Gos reference corpus of Spoken Slovenian (Zwitter Vitez et al., 2013), a collection of transcribed audio recordings of spontaneous speech in different everyday situations, in both public (TV and radio shows, school lessons, academic lectures etc.) and private settings (work meetings, services, conversations between friends and family etc.).

The SST treebank currently amounts to 29,488 tokens (3,188 utterances), which include both lexical tokens (words) and tokens signalling other types of verbal phenomena, such as filled pauses (fillers) and unfinished words, as well as some basic markers of prosody and extralinguistic speech events. The original segmentation, tokenization and spelling principles described by Verdonik et al. (2013) have also been inherited by SST. Among the two types of Gos transcriptions (pronunciation-based and normalized spelling, both in lowercase only), subsequent manual annotations in SST have been performed on top of normalized transcriptions.

For syntactic annotation of the transcripts, unavailable in Gos, the SST treebank adopted the Universal Dependencies annotation scheme due to its high degree of interoperability across different grammatical frameworks, languages and modalities. In this original application of the UD scheme to spoken language transcripts, several modifications of the scheme were implemented to accom-

modate the syntactic particularities in speech, either by extending the scope of application of existing universal labels (e.g. using *punct* for labeling markers of prosody) or introducing new speech-specific sub-labels (e.g. *discourse:filler* for annotation of hesitation sounds). In subsequent comparison of the SST treebank with the written SSJ Slovenian UD treebank (Dobrovoljc et al., 2017), Dobrovoljc and Nivre (2016) observed several syntactic differences between the two modalities, as also illustrated in Figure 1.

4 Experiment setup

4.1 Parsing systems and evaluation

To enable system-independent generalizations, two parsing systems were selected, UDPipe 1.2 (Straka and Straková, 2017) and Stanford (Dozat et al., 2017), covering the two most common parsing approaches, transition-based and graph-based parsing (Aho and Ullman, 1972), respectively. UDPipe 1.2 is a trainable pipeline for sentence segmentation, tokenization, POS tagging, lemmatization and dependency parsing. It represents an improved version of the UDPipe 1.1 (used as a baseline system in the CONLL-2017 Shared Task (Zeman et al., 2017)) and finished as the 8th best system out of 33 systems participating in the task.

A single-layer bidirectional GRU network together with a case insensitive dictionary and a set of automatically generated suffix rules are used for sentence segmentation and tokenization. The part of speech tagging module consists of a guesser, which generates several universal part of speech (XPOS), language-specific part of speech (UPOS), and morphological feature list (FEATS) tag triplets for each word according to its last four characters. These are given as an input to an averaged perceptron tagger (Straka et al., 2016) to perform the final disambiguation on the generated tags. Transition-based dependency parser is based on a shallow neural network with one hidden layer and without any recurrent connections, making it one of the fastest parsers in the CONLL-2017 Shared Task. We used the default parameter configuration of ten training iterations and a hidden layer of size 200 for training all the models.

Stanford parser is a neural graph-based parser (McDonald et al., 2005) capable of leveraging word and character based information in order to produce part of speech tags and labeled dependency parses from segmented and tokenized sequences of words. Its architecture is based on a deep biaffine neural dependency parser presented by (Dozat and Manning, 2016), which uses a multilayer bidirectional LSTM network to produce vector representations for each word. These representations are used as an input to a stack of biaffine classifiers capable of producing the most probable UD tree for every sentence and the most probable part of speech tag for every word. The system was ranked first according to all five relevant criteria in the CONLL-2017 Shared Task. Same hyperparameter configuration was used as reported in (Dozat et al., 2017) with every model trained for 30,000 training steps. For the parameters values that were not explicitly mentioned in (Dozat et al., 2017), default values were used.

For both parsers, no additional fine-tuning was performed for any specific data set, in order to minimize the influence of training procedure on the parser's performance for different data preprocessing techniques, especially given that no development data has been released for the small SST treebank.

For evaluation, we used the official CoNLL-ST-2017 evaluation script (Zeman et al., 2017) to calculate the standard labeled attachments score (LAS), i.e. the percentage of nodes with correctly assigned reference to parent node, including the label (type) of relation. For baseline experiments involving parsing of raw transcriptions (see Section 4.2), for which the number of nodes in gold-standard annotation and in the system output might vary, the F_1 LAS score, marking the harmonic mean of precision an recall LAS scores, was used instead.

4.2 Baseline

Prior to experiments involving different data modifications, both parsing systems were evaluated on the written SSJ and spoken SST Slovenian treebanks, released as part of UD version 2.2 (Nivre et al., 2018).[1] The evaluation was performed both for parsing raw text (i.e. automatic tokenization, segmentation, morphological annotation and dependency tree generation) and parsing

[1]Note that the SST released as part of UD v2.2 involves a different splitting of utterances into training and test tests as in UD v2.0, which should be taken into account when comparing our results to the results reported in the CoNLL 2017 Shared Task.

	UDPipe				Stanford			
	Parsing raw text							
Treebank	Sents	UPOS	UAS	LAS	Sents	UPOS	UAS	LAS
sst	20.35	88.32	52.49	45.47	20.35	93.21	60.35	54.00
ssj	76.49	94.59	79.90	76.32	76.49	96.32	87.50	85.02
ssj_20k	76.42	89.88	71.79	66.40	76.42	94.61	82.60	78.60
	Dependency parsing only							
Treebank	Sents	UPOS	UAS	LAS	Sents	UPOS	UAS	LAS
sst	100	100	74.66	69.13	100	100	77.58	72.52
ssj	100	100	90.16	88.41	100	100	95.63	94.52
ssj_20k	100	100	86.69	84.21	100	100	91.93	89.60

Table 1: UDPipe and Stanford sentence segmentation (Sents), part-of-speech tagging (UPOS), unlabelled (UAS) and labelled attachment (LAS) F_1 scores on the spoken SST and written SSJ Slovenian UD treebanks for parsing raw text, and for parsing texts with gold-standard tokenization, segmentation and tagging information.

gold-standard annotations (i.e. dependency parsing only). For Stanford parser, which only produces tags and dependency labels, the UDPipe tokenization and segmentation output was used as input.

The results displayed in Table 1 (Parsing raw text) confirm the difficulty of parsing spoken language transcriptions, given that both UDPipe and Stanford systems perform significantly worse on the spoken SST treebank in comparison with the written SSJ treebank, with the difference in LAS F_1 score amounting to 30.85 or 31.02 percentage points, respectively. These numbers decrease if we neutralize the important difference in treebank sizes - with 140.670 training set tokens for the written SSJ and 29.488 tokens for the spoken SST - by training the written model on a comparable subset of SSJ training data (20.000 tokens), however, the difference between the two modalities remains evident.

A subsequent comparison of results in dependency parsing only (Table 1, Dependency parsing only) reveals that a large share of parsing mistakes can be attributed to difficulties in lower-level processing, in particular utterance segmentation (with an F_1 score of 20.35),[2] as spoken language parsing performance increases to the (baseline) LAS score of 69.13 and 72.52 for the UDPipe and Stanford parser, respectively. Consequently, the actual difference between written and spoken language parsing reduces to approximately 15-17 percentage points, if based on the same amount of training data.

In order to prevent the dependency parsing experiments in this paper being influenced by the performance of systems responsible for producing other levels of linguistic annotation, the experiments set out in the continuation of this paper focus on evaluation of gold-standard dependency parsing only.

4.3 Data modifications

Given the observed difference in parsing spoken and written language for both parsing systems, several automated modifications of the data featured in the parsing pipeline have been introduced, to investigate the influence of different factors on spoken language parsing performance.

4.3.1 Modifications of training data type

Although the relationship between written and spoken language has often been portrayed as a domain-specific dichotomy, both modalities form part of the same language continuum, encouraging further investigations of cross-modal model transfers. In the first line of experiments, we thus conducted experiments on evaluation of spoken language parsing by training on spoken (`sst`) and written (`ssj`) data alone, as well as on the combination of both (`sst+ssj`). Given that the transcriptions in the SST treebank are written in lowercase only and do not include any written-like punctuation, two additional models excluding these features were generated for the written treebank (`ssj_lc` and `ssj_no-punct`) to neutralize the differences in writing system conventions

[2]Note that the low segmentation score is not specific to UDPipe, but to state-of-the-art parsing systems in general, as none of the 33 systems competing in the CoNLL 2017 Shared Task managed to achieve a significantly better result in SST treebank segmentation: http://universaldependencies.org/conll17/results-sentences.html.

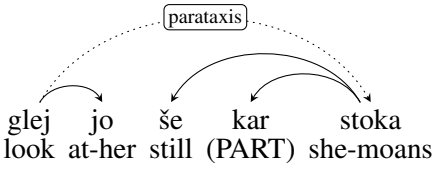
for both modalities.

4.3.2 Modifications of speech transcription

The second line of experiments investigates the role of spoken language transcription conventions for the most common speech-specific phenomena, by introducing various automatically converted versions of the SST treebank (both training and testing data).

Spelling: For word form spelling, the original normalized spelling compliant with standard orthography was replaced by pronunciation-based spelling (`sst_pron-spell`), reflecting the regional and colloquial pronunciation variation (e.g. the replacement of the standard pronominal word form *jaz* "I" by pronunciation-based word forms *jz*, *jaz*, *jst*, *jez*, *jes*, *ja* etc.).

Segmentation: Inheriting the manual segmentation of the reference Gos corpus, sentences (utterances) in SST correspond to "semantically, syntactically and acoustically delimited units" (Verdonik et al., 2013). As such, the utterance segmentation heavily depends on subjective interpretations of what is the basic functional unit in speech, in line with the multitude of existing segmentation approaches, based on syntax, semantics, prosody, or their various combinations (Degand and Simon, 2009). To evaluate parsing performance for alternative types of segmentation, based on a more objective set of criteria, two additional SST segmentations were created. In the minimally segmented version of the SST treebank (`sst_min-segm`), utterances involving two or more clauses joined by a *parataxis* relation (denoting a loose inter-clausal connections without explicit coordination, subordination, or argument relation) have been split into separate syntactic trees (clauses), as illustrated in the example below (Figure 2).

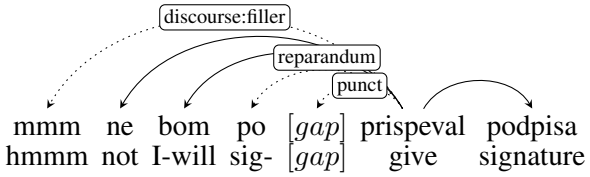


(look at her she's still moaning)

Figure 2: Splitting utterances by *parataxis*.

Vice versa, the maximally segmented SST version (`sst_max-segm`) includes utterances corresponding to entire turns (i.e. units of speech by one speaker), in which neighbouring utterances by a speaker have been joined into a single syntactic tree via the *parataxis* relation.

Disfluencies: Following the traditional approaches to spoken language processing, the `sst_no-disfl` SST treebank version marks the removal of disfluencies, namely filled pauses, such as *eee*, *aaa*, *mmm* (labeled as *discourse:filler*), overridden disfluencies, such as repetitions, substitutions or reformulations (labeled as *reparandum*), and *[gap]* markers, co-occurring with unfinished or incomprehensible speech fragments (Figure 3).



(uhm I will not sig- [gap] give my signature)

Figure 3: Removal of disfluencies.

Similar to structurally 'redundant' phenomena described above, the `sst_no-discourse` version of the SST treebank excludes syntactically peripheral speech-specific lexical phenomena, annotated as *discourse*, *discourse:filler* or *parataxis:discourse*, such as interjections (*aha* "uh-huh"), response tokens (*ja* "yes"), expressions of politeness (*adijo* "bye"), as well as clausal and non-clausal discourse markers (*no* "well", *mislim* "I think").

Prosody: Although the SST treebank lacks phonetic transcription, some basic prosodic information is provided through specific tokens denoting exclamation or interrogation intonation, silent pauses, non-turn taking speaker interruptions, vocal sounds (e.g. laughing, sighing, yawning) and non-vocal sounds (e.g. applauding, ringing). In contrast to the original SST treebank, in which these nodes were considered as regular nodes of dependency trees (labeled as *punct*), prosodic markers have been excluded from the `sst_no-pros` version of the treebank.

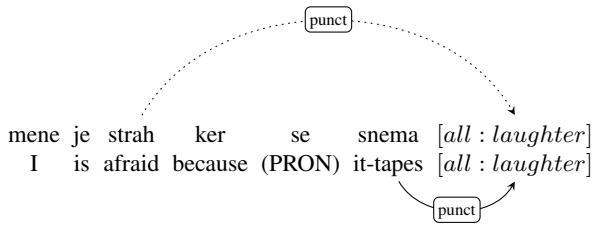
4.3.3 Modifications of UD annotation

Given that the SST treebank was the first spoken treebank to be annotated using the UD annotation scheme, the UD annotation principles for speech-specific phenomena set out in Dobrovoljc and Nivre (2016) have not yet been evaluated within a wider community. To propose potential

future improvements of the UD annotation guidelines for spoken language phenomena, the third set of SST modifications involved alternations of selected speech-specific UD representations.

Extensions: The SST treebank introduced five new subtypes of existing UD relations to annotate filled pauses (*discourse:filler*), clausal repairs (*parataxis:restart*), clausal discourse markers (*parataxis:discourse*) and general extenders (*conj:extend*). In the `sst_no-extensions` version of the treebank, these extensions have been replaced by their universal counterparts (i.e. *discourse*, *parataxis* and *conj*).

Head attachment: For syntactic relations, such as *discourse* or *punct*, which are not directly linked to the predicate-driven structure of the sentence, the choice of the head node to which they attach to is not necessarily a straightforward task. The original SST treebank followed the general UD principle of attaching such nodes to the highest node preserving projectivity, typically the head of the most relevant nearby clause or clause argument. To evaluate the impact of such high attachment principle on parsing performance, an alternative robust attachment has been implemented for two categories with the weakest semantic connection to the head, filled pauses (`sst_discourse:filler`) and prosodic markers (`sst_punct`), attaching these nodes to the nearest preceding node instead, regardless of its syntactic role, as illustrated in Figure 4.

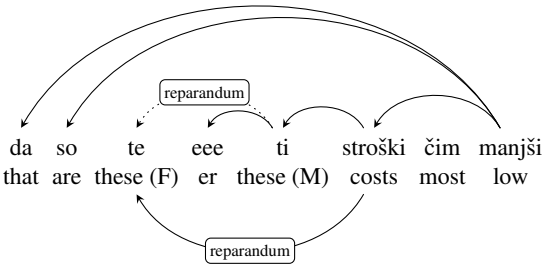


(I am afraid because it's being taped)

Figure 4: Change of head for prosody markers.

For the reparandum relation, which currently denotes a relation between the edited unit (the reparandum) and its repair, the opposite principle was implemented in `sst_reparandum`, by attaching the reparandum to the head of its repair, i.e. to the node it would attach to had it not been for the repair (Figure 5).

Following a similar higher-attachment principle, the *parataxis:restart* relation, used for

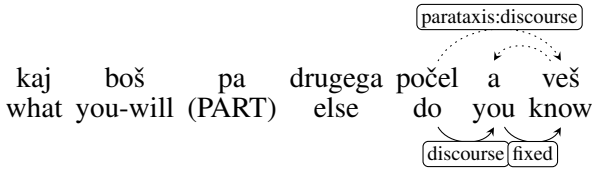


(so that these costs are as low as possible)

Figure 5: Change of head for *reparandum*.

annotation of sentences replacing an abandoned preceding clause, has been modified in `sst_parataxis:restart` so as to span from the root node instead of the more or less randomly positioned head of the unfinished clause.

Clausal discourse markers: In the original SST treebank, clausal discourse markers (e.g. *ne vem* "I don't know", (*a*) *veš* "you know", *glej* "listen") have been labeled as parataxis (specifically, the *parataxis:discourse* extension), in line with other types of sentential parentheticals. Given the distinct distributional characteristics of these expressions (limited list, high frequency) and similar syntactic behaviour to non-clausal discourse markers (no dependents, both peripheral and clause-medial positions), their label has been changed to *discourse* in the `sst_parataxis:discourse` version of the treebank. For multi-word clausal markers, the *fixed* label was also introduced to annotate the internal structure of this highly grammaticized clauses (Figure 6.



(what else can you do you know)

Figure 6: Change of annotation for clausal discourse markers.

5 Results

Table 2 gives LAS evaluation of both parsing systems for each data modification described in Section 4.3 above, including the baseline results for training and parsing on the original SST treebank

	Model	UDPipe	Stanford
Training data			
1	sst (= baseline)	69.13	72.52
2	ssj+sst	68.53	**77.38**
3	ssj_no-punct	57.40	62.57
4	ssj	55.76	62.08
5	ssj_lc	55.61	61.99
Transcriptions			
6	sst_min-segm	**74.89**	**78.31**
7	sst_no-disfl	**71.47**	**74.77**
8	sst_no-discourse	**70.73**	**75.47**
9	sst_no-pros	68.70	71.78
10	sst_pron-spell	67.52	71.64
11	sst_max-segm	63.93	68.13
Annotations			
12	sst_punct	**71.32**	**73.65**
13	sst_discourse:filler	69.13	**72.85**
14	sst_parataxis:restart	68.53	71.95
15	sst_no-new-ext.	68.45	**73.05**
16	sst_reparandum	68.41	**72.81**
17	sst_parataxis:disc.	68.32	72.35
Best combination			
18	sst_6-7-8-12	**79.58**	N/A
19	sst_6-7-8-12-15	N/A	**87.35**

Table 2: LAS on the Spoken Slovenian Treebank (sst) for different types of training data, transcription and annotation modifications. Improvements of the baseline are marked in bold.

(see Section 4.2).

When evaluating the impact of different types of training data on the original SST parsing, both parsers give significantly poorer results than the baseline sst model if trained on the written SSJ treebank alone (ssj), which clearly demonstrates the importance of (scarce) spoken language treebanks for spoken language processing. In addition, no significant improvement is gained if the written data is modified so as to exclude punctuation (ssj_no-punct) or perform lowercasing (ssj_lc), which even worsens the results. Somewhat surprisingly, no definite conclusion can be drawn on the joint training model based on both spoken and written data (sst+ssj), as the parsers give significantly different results: while Stanford parser substantially outperforms the baseline result when adding written data to the model (similar to the findings by Caines et al. (2017)), this addition has a negative affect on UDPipe. This could be explained by the fact that global, exhaustive, graph-based parsing systems are more capable of leveraging the richer contextual information gained with a larger train set in comparison with local, greedy, transition-based systems (McDonald and Nivre, 2007).

The results of the second set of experiments, in which LAS was evaluated for different types of spoken language transcriptions, confirm that parsing performance varies with different approaches to transcribing speech-specific phenomena. As expected, both systems achieve significantly better results if parsing is performed on shorter utterances (sst_min-segm). On the other hand, a similar LAS drop-off interval is identified for parsing full speaker turns (sst_max-segm). These results confirm the initial observations in Section 4.2 that speech segmentation is the key bottleneck in the spoken language dependency parsing pipeline. Nevertheless, it is encouraging to observe that even the absence of any internal segmentation of (easily identifiable) speaker turns returns moderate parsing results.

As has already been reported in related work, parsing performance also increases if spoken data is removed of its most prominent syntactic structures, such as disfluencies, discourse markers and fillers. Interestingly, for Stanford parser, the removal of discourse markers (sst_no-discourse) is even more beneficial than the removal of seemingly less predictable false starts, repairs and other disfluencies (sst_no-disfl). On the contrary, the removal of prosody markers (sst_no-pros) damages the baseline results for both parsers, suggesting that the presence of these markers might even contribute to parsing accuracy for certain types of constructions given their punctuation-like function in speech.

As for spelling, the results on the treebank based on pronunciation-based word spelling (sst_pron-spell) support our initial hypothesis that the multiplication of token types damages parser performance, yet not to a great extent. This could be explained by the fact that token pronunciation information can sometimes help with syntactic disambiguation of the word form in context, if a certain word form pronunciation is only associated with a specific syntactic role (e.g. the colloquial pronunciation *tko da* of the discourse connective *tako da* "so that" that does not occur with other syntactic roles of this lexical string).

No definite conclusion can be drawn from the parsing results for different alternations of speech-specific UD annotations, as the results vary by parsing system and by the types of UD modification. While both systems benefit from an alternative attachment of prosodic markers to their nearest preceding token (`sst_punct`),[3] and prefer the current labeling and attachment principles for clausal repairs (`sst_parataxis:restart`) and clausal discourse markers (`parataxis:discourse`), the effect of other changes seems to be system-dependent. What is more, none of the changes in UD representations seem to affect the parsing performance to a great extent, which suggests that the original UD adaptations for speech-specific phenomena, applied to the Spoken Slovenian Treebank, represent a reasonable starting point for future applications of the scheme to spoken language data.

Finally, all transcription and annotation variables that were shown to improve spoken language LAS for each of the parsing systems, have been joined into a single representation, i.e. a treebank with new, syntax-bound utterance segmentation, excluding disfluencies and discourse elements, and a change in prosody-marker-attachment (UDPipe), as well as a change in filler-attachment and addition of written parsing model (Stanford).[4] Both UDPipe and Stanford achieved substantially higher LAS scores for their best-fitting combination than the original SST baseline model (`sst`), i.e. 79.58 and 87.35, respectively, moving the SST parsing performance much closer to the performance achieved on its same-size written counterpart (`ssj_20k`, Table 1), with the gap narrowing to 4.63 for UDPipe and 2.25 for Stanford. This confirms that the speech-specific phenomena outlined in this paper are indeed the most important phenomena affecting spoken language processing scores. Nevertheless, the remaining gap between the two modalities encourages further data-based investigations into the complexity of spoken language syntax, which evidently reaches beyond the prototypical structural and pragmatic phenomena set forward in this paper and the literature in general.

[3] Note that the `sst_punct` results should be interpreted with caution, as a brief analysis into the *punct*-related parsing errors on the original SST treebank revealed a substantial amount of (incorrect) non-projective attachments of the [*gap*] marker indicating speech fragments. This issue should be resolved in future releases of the SST treebank.

[4] Modifications set out in 13 (`sst_discourse:filler`) and 16 (`sst_reparandum`) that have also increased Stanford parser performance, are not applicable to the Stanford best-combination representation, since discourse fillers and repairs have already been removed by modifications set out in 7 (`sst_no-disfl`) and 8 (`sst_no-discourse`).

6 Conclusion and Future Work

In this paper, we have investigated which speech-specific phenomena are responsible for below optimal parsing performance of state-of-the-art parsing systems. Several experiments on Spoken Slovenian Treebank involving training data and treebank modifications were performed in order to identify and narrow the gap between the performances on spoken and written language data. The results show that besides disfluencies, the most common phenomena addressed in related work, segmentation of clauses without explicit lexical connection is also an important factor in low parsing performance. In addition to that, our results suggest that for graph-based parsing systems, such as Stanford parser, spoken language parsing should be performed by joint modelling of both spoken and written data excluding punctuation.

Other aspects of spoken data representation, such as the choice of spelling, the presence of basic prosodic markers and the syntactic annotation principles seem less crucial for the overall parser performance. It has to be emphasized, however, that the UD annotation modifications set forward in this paper represent only a few selected transformations involving labeling and attachment, whereas many other are also possible, in particular experiments involving enhanced representations (Schuster and Manning, 2016).

These findings suggest several lines of future work. For the SST treebank in particular and spoken language treebanks in general, it is essential to increase the size of annotated data and reconsider the existing transcription and annotation principles to better address the difficulties in spoken language segmentation and disfluency detection. Particularly in relation to the latter, our results should be evaluated against recent speech-specific parsing systems references in Section 2, as well as other state-of-the-art dependency parsers. A promising line of future work has also been suggested in related work on other types of noisy data (Blodgett et al., 2018), employing a variety of cross-domain strategies for improving parsing with little

in-domain data.

Our primary direction of future work, however, involves an in-depth evaluation of parsing performance for individual dependency relations, to determine how the modifications presented in this paper affect specific constructions, and to overcome the prevailing approaches to spoken language parsing that tend to over-generalize the syntax of speech.

References

Alfred V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation, and Compiling*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Su Lin Blodgett, Johnny Wei, and Brendan O'Connor. 2018. Twitter Universal Dependency parsing for African-American and Mainstream American English. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1415–1425. Association for Computational Linguistics.

Andrew Caines, Michael McCarthy, and Paula Buttery. 2017. Parsing transcripts of speech. In *Proceedings of the Workshop on Speech-Centric Natural Language Processing*, pages 27–36. Association for Computational Linguistics.

Eugene Charniak and Mark Johnson. 2001. Edit detection and parsing for transcribed speech. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*, NAACL '01, pages 1–9, Stroudsburg, PA, USA. Association for Computational Linguistics.

Liesbeth Degand and Anne Catherine Simon. 2009. On identifying basic discourse units in speech: theoretical and empirical issues. *Discours*, 4.

Kaja Dobrovoljc, Tomaz Erjavec, and Simon Krek. 2017. The Universal Dependencies Treebank for Slovenian. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing, BSNLP@EACL 2017, Valencia, Spain, April 4, 2017*, pages 33–38.

Kaja Dobrovoljc and Joakim Nivre. 2016. The Universal Dependencies Treebank of Spoken Slovenian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Timothy Dozat and Christopher D. Manning. 2016. Deep biaffine attention for neural dependency parsing. *CoRR*, abs/1611.01734.

Timothy Dozat, Peng Qi, and Christopher D. Manning. 2017. Stanford's graph-based neural dependency parser at the CoNLL 2017 Shared Task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30, Vancouver, Canada. Association for Computational Linguistics.

John J. Godfrey, Edward C. Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *Proceedings of the 1992 IEEE International Conference on Acoustics, Speech and Signal Processing - Volume 1*, ICASSP'92, pages 517–520, Washington, DC, USA. IEEE Computer Society.

Erhard W. Hinrichs, Julia Bartels, Yasuhiro Kawata, Valia Kordoni, and Heike Telljohann. 2000. The Tübingen treebanks for spoken German, English, and Japanese. In Wolfgang Wahlster, editor, *Verbmobil: Foundations of Speech-to-Speech Translation*, Artificial Intelligence, pages 550–574. Springer Berlin Heidelberg.

Matthew Honnibal, Yoav Goldberg, and Mark Johnson. 2013. A non-monotonic arc-eager transition system for dependency parsing. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 163–172, Sofia, Bulgaria. Association for Computational Linguistics.

Matthew Honnibal and Mark Johnson. 2014. Joint incremental disfluency detection and dependency parsing. *Transactions of the Association for Computational Linguistics*, 2(1):131–142.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378. Association for Computational Linguistics.

Fredrik Jørgensen. 2007. The effects of disfluency detection in parsing spoken language. In *Proceedings of the 16th Nordic Conference of Computational Linguistics NODALIDA-2007*, pages 240–244.

Jeremy G Kahn, Matthew Lease, Eugene Charniak, Mark Johnson, and Mari Ostendorf. 2005. Effective use of prosody in parsing conversational speech. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 233–240. Association for Computational Linguistics.

Anne Lacheret, Sylvain Kahane, Julie Beliao, Anne Dister, Kim Gerdes, Jean-Philippe Goldman, Nicolas Obin, Paola Pietrandrea, and Atanas Tchobanov. 2014. Rhapsodie: a prosodic-syntactic treebank for spoken French. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 295–301, Reykjavik, Iceland. European Language Resources Association (ELRA).

Matthew Lease, Mark Johnson, and Eugene Charniak. 2006. Recognizing disfluencies in conversational speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5):1566–1573.

Yang Liu, Elizabeth Shriberg, Andreas Stolcke, Dustin Hillard, Mari Ostendorf, and Mary Harper. 2006. Enriching speech recognition with automatic detection of sentence boundaries and disfluencies. *IEEE Transactions on audio, speech, and language processing*, 14(5):1526–1540.

Ryan McDonald and Joakim Nivre. 2007. Characterizing the errors of data-driven dependency parsing models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 523–530. Association for Computational Linguistics.

Alexis Nasr, Frederic Bechet, Benoit Favre, Thierry Bazillon, Jose Deulofeu, and andre Valli. 2014. Automatically enriching spoken corpora with syntactic information for linguistic studies. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Joakim Nivre. 2015. Towards a universal grammar for natural language processing. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 9041 of *Lecture Notes in Computer Science*, pages 3–16. Springer International Publishing.

Joakim Nivre, Jens Nilsson, and Johan Hall. 2006. Talbanken05: A Swedish treebank with phrase structure and dependency annotation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, pages 1392–1395.

Joakim Nivre et al. 2018. Universal Dependencies 2.2. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (UFAL), Faculty of Mathematics and Physics, Charles University.

Mohammad Sadegh Rasooli and Joel Tetreault. 2013. Joint parsing and disfluency detection in linear time. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 124–129, Seattle, Washington, USA. Association for Computational Linguistics.

Sebastian Schuster and Christopher D. Manning. 2016. Enhanced english universal dependencies: An improved representation for natural language understanding tasks. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Milan Straka, Jan Hajic, and Jana Straková. 2016. Udpipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Milan Straka and Jana Straková. 2017. Tokenizing, pos tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.

Trang Tran, Shubham Toshniwal, Mohit Bansal, Kevin Gimpel, Karen Livescu, and Mari Ostendorf. 2017. Joint modeling of text and acoustic-prosodic cues for neural parsing. *CoRR*, abs/1704.07287.

Darinka Verdonik, Iztok Kosem, Ana Zwitter Vitez, Simon Krek, and Marko Stabej. 2013. Compilation, transcription and usage of a reference speech corpus: the case of the Slovene corpus GOS. *Language Resources and Evaluation*, 47(4):1031–1048.

Ton van der Wouden, Heleen Hoekstra, Michael Moortgat, Bram Renmans, and Ineke Schuurman. 2002. Syntactic analysis in the Spoken Dutch Corpus (CGN). In *Proceedings of the Third International Conference on Language Resources and Evaluation, LREC 2002, May 29-31, 2002, Las Palmas, Canary Islands, Spain*.

Masashi Yoshikawa, Hiroyuki Shindo, and Yuji Matsumoto. 2016. Joint transition-based dependency parsing and disfluency detection for automatic speech recognition texts. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1036–1041.

Daniel Zeman et al. 2017. CoNLL 2017 Shared Task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada. Association for Computational Linguistics.

Ana Zwitter Vitez, Jana Zemljarič Miklavčič, Simon Krek, Marko Stabej, and Tomaž Erjavec. 2013. Spoken corpus Gos 1.0. Slovenian language resource repository CLARIN.SI.