Genres, Parsers, and BERT: The Interaction Between Parsers and BERT Models in Cross-Genre Constituency Parsing in English and Swedish

Daniel Dakota
Uppsala University
Department of Linguistics
ddakota@lingfil.uu.se

Abstract

Genre and domain are often used interchangeably, but are two different properties of a text. Successful parser adaptation requires both cross-domain and cross-genre sensitivity (Rehbein and Bildhauer, 2017). While the impact domain differences have on parser performance degradation is more easily measurable in respect to lexical differences, impact of genre differences can be more nuanced. With the predominance of pre-trained language models (LMs; e.g. BERT (Devlin et al., 2019)), there are now additional complexities in developing cross-genre sensitive models due to the infusion of linguistic characteristics derived from, usually, a third genre. We perform a systematic set of experiments using two neural constituency parsers to examine how different parsers behave in combination with different BERT models with varying source and target genres in English and Swedish. We find that there is extensive difficulty in predicting the best source due to the complex interactions between genres, parsers, and LMs. Additionally, the influence of the data used to derive the underlying BERT model heavily influences how best to create more robust and effective cross-genre parsing models.

1 Introduction

The performance degradation of models trained on one data set when used on another has been well established (Gildea, 2001; Petrov and Klein, 2007). However, how we define the source of the problem (e.g. *out-of-domain differences*) is problematic. Within domain adaption, even the term *domain* is incredibly loosely defined (Ramponi and Plank, 2020). This has allowed conflating several different properties of texts, as such properties can be difficult to distinguish given some of their inherent overlap. Relevant for this work is how we define the distinction between *genre* and *domain*.

We use Falkenjack et al. (2016) as a template and define *genre* dealing with more *abstract* and *linguistic characteristics* used within a text (Biber and Conrad, 2009). *Domain*, however, is more about the topics and content words used. Much parsing work has actively focused on handling domain differences, such as reducing lexical gap issues between target and source domains (e.g. Candito et al. (2011)), while explicit handling of genre differences is not as heavily researched, nor as well understood.

While much parsing literature uses the terms interchangeably (Rehbein and Bildhauer, 2017), they are not, however, identical concepts. By doing so we are are not effectively identifying which *out-of-domain* differences should be contributed more to out of *genre* or *domain* difference. For example, Wikipedia articles are written in a style following more of an encyclopedia. Pages on medicine and languages may contain very different vocabularies, but the linguistic characteristics are most likely similar given the encyclopedic style of writing. However, responses in a forum on medical advice may share a large amount of vocabulary overlap with Wikipedia medical pages, but would share considerably less linguistic structure given the dialogue nature of a forum (e.g. more interrogative sentences).

A treebank (in our case constituency treebanks) often contains many noticeable domains, but it is harder to gauge how many distinct genres are present. Sometimes they are explicitly marked in the annotation (Candito and Seddah, 2012; Telljohann et al., 2015), while others explicitly separate out the different genres out (McDonald et al., 2011; Adesam et al., 2015). Yet a treebank is often times a concatenation of various texts that may ultimately represent slightly different linguistic characteristics (even if annotated strictly on newspaper).

Domain differences often result in a high lexi-

Proceedings of the Second Workshop on Domain Adaptation for NLP, pages 59–71

cal divergence. The best performing chart-based grammar-based constituency parsers were predominantly unelexicalized (Petrov and Klein, 2007), which helped reduce issues with lexical differences. However, current state-of-the-art neural span-based chart-based parsers have substantially changed this paradigm. With the use of lexical embeddings, there is now a great deal of lexicalization within the modeling architecture to an extent that was not seen before, with some state-of-the-art parsers not utilizing POS tags (Zhang et al., 2020). The use of characters and subtoken information has been shown to be beneficial in reducing lexical sparsity issues (Vania et al., 2018), which ultimately reduces domain difference disparities. However, what impact this degree of lexical contextualization has on cross-genre parsing remains unclear.

Additionally, the use of language models (LMs), such as BERT (Devlin et al., 2019), to derive contextualized embeddings presents yet another variable in selecting the best source. Given that LMs are derived on large, unannotated texts, they implicitly capture various linguistic properties of the these texts (Tenney et al., 2019a) which they they infuse into the parser via contextualized embeddings. How embeddings derived from, often, a single genre behave as a bridge between two, often, other genres presents an interesting issue.

We are interested in examining the following in cross-genre experiments:

1. What preferences do different parsing architectures exhibit?

2. What interactions do different BERT models demonstrate?

3. What behaviors are exhibited across languages?

2 Related Work

Most parsing work[1] in parser adaptation has been more explicitly focused on issues of domain differences. Early techniques focused on selecting optimal source data to boost a target set (Plank and van Noord, 2011; McDonald et al., 2011) or parameter and model optimization to handle both general and domain specific features (Daumé III, 2007; Kim et al., 2016). Both delexicalized (Rosa and Žabokrtský, 2015) and lexicalized (Falenska and Çetinoğlu, 2017) similarity metrics have shown the ability to select optimal source data.

More recent work has been focused on creating domain specific embeddings. The use of domain embeddings in Chinese dependency parsing by Li et al. (2019) built on the previous research by Stymne et al. (2018). Both showed domain and treebank specific embeddings respectively yielded better performance over direct treebank concatenation, as this allows for the capturing of domain specific and general features. Results were further improved upon with adversarial methods and BERT fine-tuning by Li et al. (2020).

Joshi et al. (2018) found that contextualized embeddings have substantially reduced the difficulty in handling lexical gap issues between domains when the target and source are syntactically similar, but then employ additional strategies to handle more syntactically dissimilar ones. Additional work by Fried et al. (2019) showed that while pretrained LMs improved parser performance over several English domains, the improvements for out-of-domain results were not relatively larger. As Rehbein and Bildhauer (2017) note, parser adaptation requires both genre and domain adaption, but that content features, such as topics, do not generalize well for genre modeling, suggesting different techniques are needed for cross-genre modeling.

While the incorporation of pre-trained LMs has become standard in many NLP tasks, understanding how different models interact with different tasks is still an area of active research. Work by Martin et al. (2020) on French shows that a smaller French specific LM model derived from more diversified source data can compete with models substantially larger across a variety of downstream NLP tasks. Specifically, tasks which showed more divergence from Wikipedia benefited the most from a mixed genre LM. The importance of source diversification is also seen in LMs for Finnish (Virtanen et al., 2019) and Chinese (Cui et al., 2020), each of which contain more text sources than simply Wikipedia. The impact of source diversification can also be seen in domain specific LMs, such as FinBERT (Liu et al., 2020), which was derived from several types of financial sources, as different financial texts are radically different stylistically.

One additional benefit by explicitly looking at specific genres is it can help further our understanding of linguistic properties of different texts, forcing us to re-evaluate earlier annotation schemes

[1] In the related work section we use the terms used in the original papers and not our definitions.

predominantly designed for an original treebank genre (Rúnarsson and Sigursson, 2020).

3 Methodology

We perform a systematic set of experiments for English and Swedish, using different neural constituency parsing architectures in combination with various BERT models to examine how this impacts cross-genre parsing. English is widely used in cross-domain and cross-genre research. Swedish, however, is not as thoroughly examined, yet possess multiple genre treebanks as well as BERT models, making it suitable for our research interests.

3.1 Parsers

We use two different neural span-based chart-based parsers, the Berkeley Neural Parser (Kitaev et al., 2019) and the SuPar Neural CRF Parser (Zhang et al., 2020).

Berkeley Neural Parser uses a self-encoder and can incorporate BERT models to generate word representations. It uses the last layer embedding of the last subtoken to represent the word.[2] It decouples predicting the optimal representation of a span (i.e. input sequence) from predicting the optimal label, requiring only that the resultant output form a valid tree. This not only removes the underlying grammars found in traditional PCFG parsers, but also direct correlations between a constituent and a label (Fried et al., 2019). A CKY (Kasami, 1965; Younger, 1967; Cocke and Schwartz, 1970) style inference algorithm is used at test time. Additionally, the parser allows the option of using POS tag prediction to be used as an auxiliary loss task (we use BNP and BNPno to represent with and without the POS loss respectively in our experiments).

SuPar Neural CRF Parser (SuPar) is a two-stage parser, that, similarly to the Berkeley parser, produces a constituent and then a label. It uses a Scalar mix (Peters et al., 2018; Tenney et al., 2019a,b) of the last four layers for each subtoken of a word. Additionally, it uses a BiLSTM encoder to compute context aware representations by employing two different MLP layers indicating both left and right word boundaries. Each candidate is scored over the two representations using a biaffine operation (Dozat and Manning, 2017), while the CKY algorithm is used when parsing to obtain the best tree.

[2]The authors note they found no difference between using the last and first subtoken.

3.2 Treebanks

We choose to experiment on two languages that contain treebanks representative of different genres, the English Webocorpus Treebank (Petrov and McDonlad, 2012) and the Koala Eukalyptus Corpus (Adesam et al., 2015).

English Webcorpus Treebank (EWT) was introduced in the 2012 shared task on Web Parsing and consists of five subareas: Yahoo answers, emails, Newsgroup texts, product reviews, and Weblog entries. The treebank follows an English Penn Treebank (Marcus et al., 1993) style annotation scheme with some additional POS tags to account for specific annotation needs, resulting in 50 POS tags and 28 phrase heads. We removed unary nodes, traces, and function labels during preprocessing.

Swedish Eukalyptus Treebank (SET) consists of: blog entries from the SIC corpus (Östling, 2013), parts of Swedish Europarl (Koehn, 2005), chapters from books, public information gathered from government and health information sites, and Wikipedia articles, and contains only 13 POS tags and 10 phrases heads. The treebank's annotation scheme is derived from the TiGer Treebank of German (Brants et al., 2004). Notably this includes discontinuous constituents, resulting in the need to uncross the branches of the extracted treebank. We follow the procedure used for TiGer, namely the transformation process proposed by Boyd (2007) using *treetools*,[3] and additionally remove all function labels.

Data Splits The EWT is traditionally used as dev and test sets for examining the out-of-domain adaptability of models developed on the English PTB (Petrov and McDonlad, 2012), and we are not aware of any standard splits for the EWT nor of standard splits for the SET. For this reason we chose to split each genre within the treebanks into approximately sequential 80/10/10 splits, with selected treebank statistics presented in Table 1. For cross-genre experiments, EWT and SET subgenres are concatenated respectively.

3.3 BERT Embeddings

We use four different embeddings in our experiments: both bert-base-multilingual-cased and

[3]https://github.com/wmaier/treetools

Treebank	Genre	Train Sent.	Tok	Tok. Typ.	Tok. Rat.	POS Trigram Rat.	Dev Sent.
EWT	Answers	2790	42428	6344	.1495	.1367	349
	Enail	3919	45488	7056	.1551	.1103	490
	Newsgroup	1909	33764	6769	.2005	.1498	238
	Reviews	3049	44414	6362	.1432	.1156	381
	Weblog	1623	35864	6420	.1790	.1414	203
SET	Blog	1050	15050	3659	.2431	.0713	100
	Europarl	640	14580	3074	.2108	.0565	79
	Public	900	16540	4540	.2745	.0589	89
	Novels	950	16188	4114	.2552	.0594	120
	Wiki	900	16031	4788	.2937	.0364	100

Table 1: Treebank statistics for EWT and SET genres with number of train sentences along with total tokens, token type ratios, and unique POS trigram ratios for training sets, as well as number of dev sentences

Treebank		Answers	Email	Newsgroup	Reviews	Weblog
EWT	Answers	0	.2679	.2622	**.2063**	.3180
	Email	.3689	0	**.3099**	.4755	.4721
	Newsgroup	.4157	.3845	0	.4881	**.2539**
	Reviews	**.2279**	.3870	.3613	0	.4082
	Weblog	.4125	.4523	**.1945**	.4738	0
		Blog	Europarl	Public	Novels	Wiki
SET	Blog	0	.4108	.5060	**.2374**	.4633
	Europarl	.4660	0	**.2074**	.2165	.2526
	Public	.4816	.2443	0	.2665	**.1494**
	Novels	**.1991**	.2388	.2705	0	.2596
	Wiki	.4197	.3382	**.1622**	.2720	0

Answers	Email	Newsgroup	Reviews	Weblog
0	.3900	.3539	**.2987**	.4416
.4820	0	**.4044**	.6236	.6194
.4038	.4215	0	.4851	**.3213**
.2978	.4926	.4390	0	.5465
.4860	.5767	**.3201**	.5475	0
Blog	Europarl	Public	Novels	Wiki
0	.4942	.4942	**.4034**	.5134
.5109	0	**.3983**	.4965	.5572
.4413	**.3930**	0	.4217	.3983
.3783	.5003	.4469	0	.4342
.4257	.5004	**.4179**	.4637	0

Table 2: KL Divergence for POS trigrams (left) and BERTbc and kbBERT Subtokens (right)

bert-base-cased (Devlin et al., 2019), bert-large-swedish-uncased,[4] and bert-base-swedish-cased (Malmsten et al., 2020).[5] All of these have an unknown number of domains present within each model.

bert-base-multilingual-cased (mBERT) was trained on 104 languages of Wikipedia with oversampling of low resource languages and under sampling of high resource languages.

bert-base-cased (BERTbc) was trained on a mixture of English Wikipedia (2,500M words) and BookCorpus (800M words; Zhu et al., 2015) with the BookCorpus containing sixteen identified book genres.[6]

bert-large-swedish-uncase (swBERT) was trained on Swedish Wikipedia (300M words).

bert-base-swedish-cased (kbBERT) was trained using newspapers (2,977 M words), government publications (117M words), legally available e-deposits (62M words),[7] internet forums (31M words), and Swedish Wikipedia (29M words).

4 Delexicalized and Subtoken Divergence

As noted in section 2, delexicalized comparisons of treebanks have been used to identify treebank similarity for source selection. An established delexicalized method is KL divergence (Kullback and Leibler, 1951) of POS trigrams (Rosa and Žabokrtský, 2015). In Table 2 we present results for KL divergence for POS trigrams with the closest similar genre in bold.[8]

Given that BERT works on a subtoken level, we additionally present the KL divergence for BERT subword tokens between genres. Each genre was tokenized using the specified BERT tokeinzer and counts were collected on subtokens. Identifying BERT subword tokens similarities provides insights into (sub)lexical level similarity, as well as how delexicalized and subword pattern with each other.[9]

The row (y-axis) is the target genre, and the columns (x-axis) are the source (e.g. in Table 2 .4660 is Europarl target Blog source).[10] We see

[4] https://github.com/af-ai-center/SweBERT

[5] https://github.com/Kungbib/swedish-bert-models

[6] We do not consider this to mean there are 16 distinct genres as we define the term, rather to note the more diversified domains, though author style would naturally influence any learned representations.

[7] Including governmental releases, books, and magazines.

[8] We follow Rosa and Žabokrtský (2015) and default the target genre to 1 in KL calculations.

[9] We note however that the two parsers do not necessarily use all the subtokens when generating embeddings.

[10] All future heat maps and tables have the same set-up.

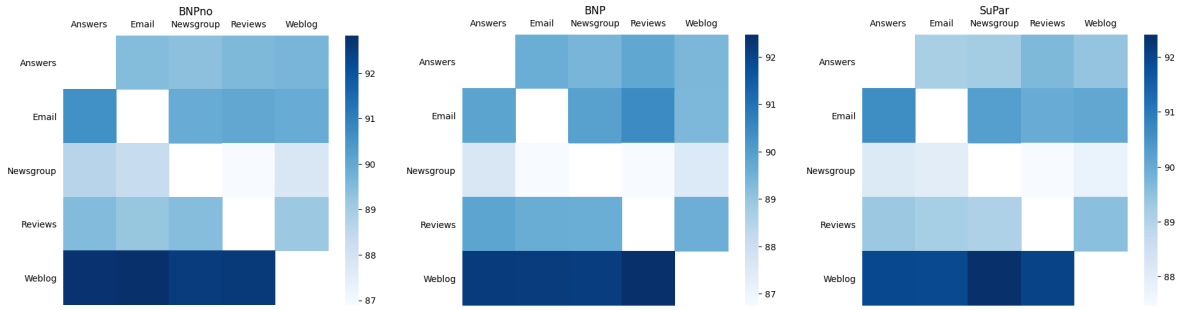

Figure 1: Heat maps for EWT with mBERT on Dev Set

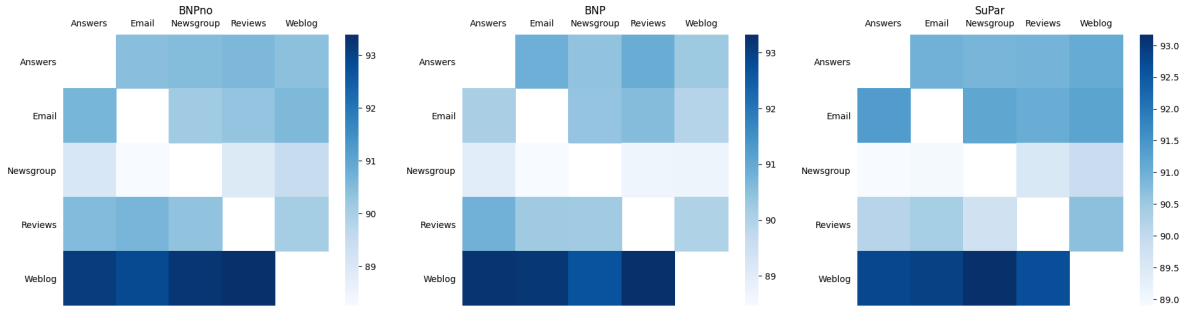

Figure 2: Heat maps for EWT with BERTbc on Dev Set

that the patterns are rather similar, with simply the degree of divergence being larger between POS trigrams and sublexical tokens. The lone exception is that on the POS level, Wiki is a better source for Public while on the subtoken level, Europarl is. We can also see that a high subword dissimilarity does not necessarily predict a proportionally high POS dissimilarity.

5 Results

5.1 EWT mBERT

Fig. 1 shows heat maps representing transfer F-scores on the dev sets for different target and source genres. Note that the diagonals are NA values, not the minimum per axis given that the diagonal represents when the target and source are the same genre. We also present a setting All[11] in which all the genres are combined in the train and indicate the absolute F-score increase over the baseline, as well as Gap which indicates the absolute increase the All setting shows compared to the best source experiment.

We see that EWT using mBert does not correlate well with the KL divergences in Table 2. There is seemingly a preference for either Answers or Reviews as the best source genre across experiments. Furthermore, no single parsing architecture can claim to be superior, as the best individual settings are quite varied across the parsers. The All setting results in the best over all performance, a trend that will continue through all results, but this is unsurprising given it has more training data across all cross-genre experiments. The individual Gap improvements show a large range of improvements, but also a lack of noted consistency about how much improvement the All setting has over the best source for each genre.

5.2 EWT BERTbc

We see noticeable improvements for all experiments when using an English specific BERT model (see Fig. 2), which is expected. However, improvements for individual settings vary greatly. In some cases, the improvements are greater than 2% absolute, while in others they are as small as .05%.

We also see a more noticeable trend to SuPar performing slightly better than BNP and BNPno in many experiments overall, particularly in the All setting, and shows consistent higher Gap increases. However, we see continued individual architectural strengths and consistency, such as that BNPno still shows strength on parsing Weblog, similar to that in Fig. 1, and BNP actually shows identical best source genres as those for mBERT.

We do, however, see more variation in source preferences for the other two architectures. For

[11] Tables containing full results are found in Appendix A.

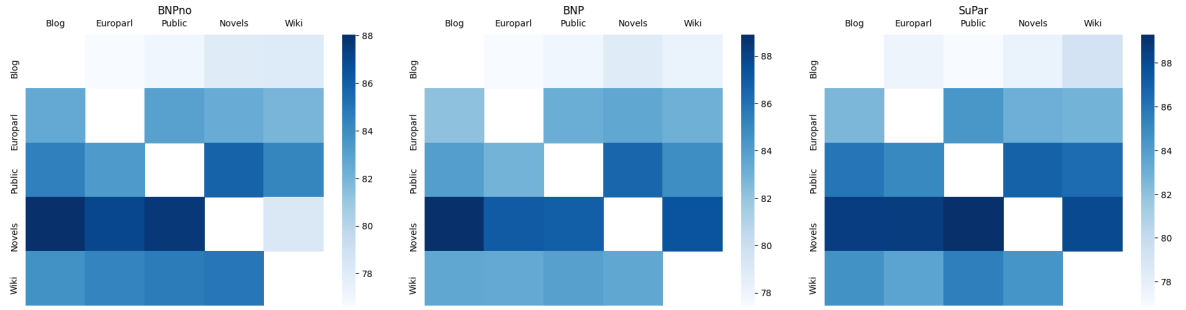

Figure 3: Heat maps for SET with mBERT on Dev Set

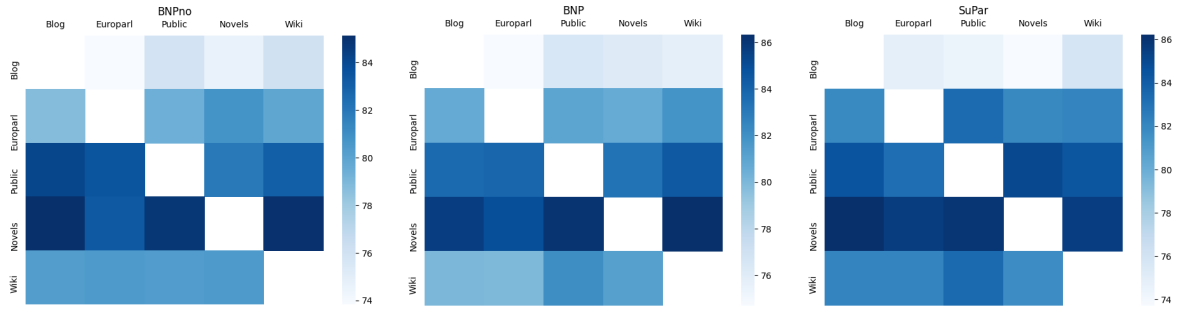

Figure 4: Heat maps for SET with swBERT on Dev Set

BNPno, Answers is no longer dominant and instead we see a great deal of variation, while for SuPar, we see a shift towards Weblog. This is particularly interesting given that Weblog is often the most dissimilar source in regards to KL.

Another interesting observation is that for both BNPno and BNP, Reviews is clearly not benefiting in the All setting as the other genres are. In fact, for BNP, we see it is is actually worse than the best source experiment.

5.3 SET mBERT

In Fig. 3 we see results using multilingual BERT on Swedish.[12] An initial observation is that SuPar performs, overall, better than both BNPno and BNP, particularly in the All setting, though there may be individual settings in which a Berkeley parser setup performs better.

In terms of individual source experiments, we see a great deal of variation intra and inter parser. While Novels is the best source for Public across three experimental setings, for all other genres, at least one of the best sources is different for that specific genre across the parsers.

For BNPno we see that Wiki is the best source for Blog, even though it is furthest in subtoken similarity, and the second furthest in POS similarity. Yet when using Wiki as a source, BNPno outperforms its BNP counterpart in every single experiment, often substantially, except in the All setting.

Europarl is seemingly a case where POS and subtoken divergences align with parser architectures in regards to lexicalization. For the BNP, Novels is preferred, which is just behind Public in POS divergence but substantially behind in subtoken divergence. However, both BNPno and SuPar prefer Public, which is by far the closest at the subtoken level, and actually perform relatively poorly using the other genres as sources.

5.4 SET swBERT

Generally, we see a decrease in performance for swBERT (Fig. 4) compared to mBERT. However, the drop is perhaps not as significant in many cases as expected, especially given the size difference of the LMs. Also remembering that swBERT is uncased, and many cased models work better, we are unsure how much this contributes to performance degradation. However, there are still several settings in which swBERT out performs mBERT. BNP also shows more volatility compared to BNPno, but we still see the trend that in the All setting, it still performs better. One interesting observation is the lack of variation in the results in the BNPno experiments using any different source for Wiki data.

[12] Tables containing full results are found in Appendix B.

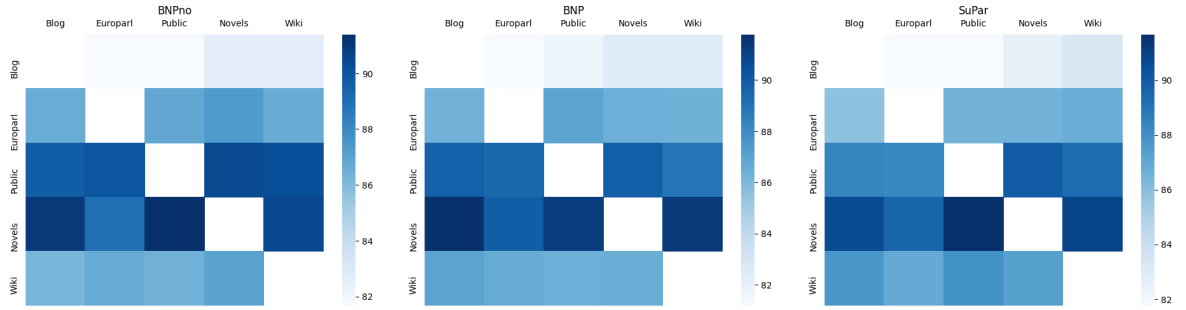

Figure 5: Heat maps for SET with kbBERT on Dev Set

Additionally, in three experiments Wiki is the best source for BNP, yet none of these sources were the best using mBERT. However, these three experiments substantially out perform their BNPno counterparts. The behavior of Blog is not intuitive, as it actually now benefits the most from one of the least similar sources in Public.

SuPar stays relatively consistent, with the only change that the best source for Novels switched from Public to Blog. However, this is actually a change to the most similar source genre, something that mBERT dispreferred.

5.5 SET kbBERT

All results for kbBERT (Fig. 5) are substantially better than both mBERT and swBERT. SuPar again shows less volatility, with Public returning as the best source for Novels, and now Wiki being the best source of Europarl.

Wiki is the best source for Blog, but we must note that for both BNPno and BNP, it is barely better than Novels, while for SuPar it is substantially better. For BNPno, we see the best performing sources are slightly different than with mBERT as now Euorparl prefers Novels and Novels prefers Public instead of Blog. For BNP, Wiki benefits the most from Blog, even though it is the most dissimilar in regards to POS divergence. Another important observation however, is that the slight performance advantage SuPar had using mBERT and swBERT over the Berkeley parsers has been somewhat reduced, and in many settings a Berkeley parser out performs SuPar again.

6 Discussion

The different parsing architectures interact differently with the underlying latent properties of the embeddings in their parsing decisions. SuPar, however, does seem to show the most consistent stable performance across all experiments, and in a majority of cases, is the best performing model.

Whether POS information is needed in neural constituency parsing is seemingly a complicated picture in terms of performance, though it has been shown to benefit certain neural dependency parsing architectures (Zhou et al., 2020). However, we can see the impact the inclusion of the POS loss has in terms of parser source preferences, as seldom were the behaviors of BNPno and BNP similar. This is to be expected, as the source of the underlying LM may have implicitly different POS distributions than either the target or source genre, and a POS loss is most likely sensitive to these differences.

BNP showed much more stable source preferences across genres and experiments, indicating how the POS task is seemingly is able to mitigate, to some degree, the influence of the LM, though whether this is positive or negative is unclear. This may indicate that embeddings derived from more mono-genre texts interact in a more consistent way when using POS information, stablizing source preferences. This is seen in both English and Swedish experiments to a degree. However, once the LM has more genre representations, this stabilizing factor no longer holds as the inherent POS distributions are most likely far more varied, as seen with kbBERT.

We can also see how other architectural choices besides the inclusion of POS information are important, as otherwise we would expect BNPno and SuPar to behave similarly, which they do not. A clear distinction is how the two parsers incorporate BERT embeddings. The choice of Scalar mixing (Liu et al., 2020; de Vries et al., 2020), embedding averages (He and Choi, 2020), and different subtoken selection (Hettiarachchi and Ranasinghe, 2020) have all shown to impact performance on NLP tasks. Another factor may be the additional word boundaries MLP layers created in SuPar's

architecture, providing more context for an individual parsing decision, making it more robust to slight variations in syntactic distributions.

The influence of the LM's genre is perhaps most seen in the Swedish Wikipedia genre experiments with swBERT and BNP seen in Fig. 4. All the sources produce in similar results when Wikipedia is the target. It may simply be that when the target genre is too similar to the genre of the LM, the impact of similar sizes of different source genres is minimized, as there now exist too much latent and explicit Wikipedia data. However, in the All setting, we see a substantial increase where now there is not only more data, but more diversity to counterbalance the Wikipedia derived LM. Additionally we see that Wikipedia is the preferred source for all genres outside of Blog for BNP, and results are substantially better than their BNPno counterparts. However, this does not hold across parsers, given that SuPar shows completely different behaviors for swBERT than the Berkeley experiments. This further emphasizes the difficulty of transferring knowledge of one parser's source preference behaviors to another.

For both languages there can be substantial deviation of the best performing source genre from the closest source genre on both a delexicalized and subtoken level. Overall gains specific sources show for an individual target source can also be incredibly inconsistent across experiments. Why this is, is further complicated by the source genres interaction with the LM. An English only BERT model yielded some improvements while a Swedish only model showed varying results depending upon the LM. This can be due to several factors. The most obvious one is due to the size of the LM, as swBERT is substantially smaller, yet it still yields results close to the much larger mBERT for Swedish. However, kbBERT is approximately the same size as BERTbc, and produces much larger absolute gains than BERTbc did for English, providing counter evidence that size is not the only factor. The difficulty in identifying the reasons is due to many interacting aspects such as higher baselines for English, treebank sizes, and annotation scheme complexity. However, kbBERT was derived from a far more diverse set of genres, many of which overlap with the SET, compared to BERTbc, which was derived from mostly two distinct genres, neither of which overlap with the EWT substantially.

Importantly, we also see the impact a LM model has on closing performance gaps between parsing architectures, as kbBERT results for the Berkeley parsers are overall more on par with SuPar. This demonstrates how interactions between three distinct genres makes optimal source selection far more difficult when using LMs and different parsers than established delexicalized approaches.

7 Conclusion

We have performed a set of detailed experiments that explored the interaction between genres, parsers, and BERT models. We have shown that the LM plays the pivotal role in successful genre-sensitive parsing within our chosen parsing architectures. In addition, we have also shown that different architectures often behave dissimilarly, making determining best sources for a specific target reliant on better understanding the underlying architectures, and not transferring direct behavior of one parser to another.

Treebanks are rather static, particularly constituency treebanks. While we have often seen incremental performance gains with every new parser, how successful we are at cross-genre parsing will, for the time being, be more related to our exploitation of various other sources and methods. LMs, for example, can be trained on vast amounts of unannotated data, allowing for the the LM to become far more sensitive to genre differences than any small treebank, especially as we have control over the creation of an LM, and less so than with a treebank.

Perhaps the most practical way to currently create genre-sensitive parsing models is to better mix distinct genres within the data used to derive the LM. The LM itself does not even have to be overtly large, rather even small mixtures of other genres and domains provides noticeable benefits (Martin et al., 2020). Future research will look to create multilingual cross-genre models that work across treebanks and genres.

Acknowledgements

The author would like to thank Yvonne Adesam and Gerlof Bouma for providing the SET, Sandra Kübler and members of the Uppsala NLP Parsing Group: Joakim Nivre, Sara Stymne, and Artur Kulmizev for their feedback, and the anonymous reviewers for their comments. The author is supported by the Swedish strategic research programme eSSENCE.

References

Yvonne Adesam, Gerlof Bouma, and Richard Johansson. 2015. Defining the eukalyptus forest – the koala treebank of Swedish. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 1–9, Vilnius, Lithuania. Linköping University Electronic Press, Sweden.

Douglas Biber and Susan Conrad. 2009. *Register, Genre, and Style*. Cambridge Textbooks in Linguistics. Cambridge University Press.

Adriane Boyd. 2007. Discontinuity Revisited: An Improved Conversion to Context-free Representations. In *Proceedings of the Linguistic Annotation Workshop*, LAW '07, pages 41–44, Prague, Czech Republic.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans. Uszkoreit. 2004. TIGER: Linguistic Interpretation of a German Corpus. *Journal of Language and Computation*, 2004 (2):597–620.

Marie Candito, Enrique Henestroza Anguiano, and Djamé Seddah. 2011. A word clustering approach to domain adaptation: Effective parsing of biomedical texts. In *Proceedings of the 12th International Conference on Parsing Technologies*, pages 37–42, Dublin, Ireland.

Marie Candito and Djamé Seddah. 2012. Le corpus sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012*, pages 321–334, Grenoble, France.

John Cocke and Jacob Schwartz. 1970. Programming Languages and Their Compilers. Technical report, Courant Institute of Mathematical Sciences, New York.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for Chinese natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 657–668, Online. Association for Computational Linguistics.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic.

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

Timothy Dozat and Christopher Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5h International Conference on Learning Representations (ICLR 2017)*, Toulon, France. Conference Track Proceedings.

Agnieszka Falenska and Özlem Çetinoğlu. 2017. Lexicalized vs. delexicalized parsing in low-resource scenarios. In *Proceedings of the 15th International Conference on Parsing Technologies*, pages 18–24, Pisa, Italy.

Johan Falkenjack, Marina Santini, and Arne Jönsson. 2016. An exploratory study on genre classification using readability features. In *Proceedings of the The Sixth Swedish Language Technology Conference (SLTC 2016)*, Umeå, Sweden.

Daniel1 Fried, Nikita Kitaev, and Dan Klein. 2019. Cross-domain generalization of neural constituency parsers. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 323–330, Florence, Italy.

Daniel Gildea. 2001. Corpus Variation and Parser Performance. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 167–202, Pittsburgh, PA.

Han He and Jinho D. Choi. 2020. Establishing strong baselines for the new decade: Sequence tagging, syntactic and semantic parsing with bert.

Hansi Hettiarachchi and Tharindu Ranasinghe. 2020. BRUMS at SemEval-2020 task 3: Contextualised embeddings for predicting the (graded) effect of context in word similarity. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 142–149, Barcelona (online). International Committee for Computational Linguistics.

Vidur Joshi, Matthew Peters, and Mark Hopkins. 2018. Extending a parser to distant domains using a few dozen partially annotated examples. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1190–1199, Melbourne, Australia.

Tadao Kasami. 1965. An Efficient Recognition and Syntax-Analysis Algorithm for Context-Free Languages. Technical report, AFCRL-65-75, Air Force Cambridge Research Laboratory.

Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2016. Frustratingly easy neural domain adaptation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 387–396, Osaka, Japan.

Nikita Kitaev, Steven Cao, and Dan Klein. 2019. Multilingual constituency parsing with self-attention and pre-training. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3499–3505, Florence, Italy.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Machine Translation Summit*, pages 79–86, Phuket, Thailand.

Solomon Kullback and Richard Leibler. 1951. On information and sufficiency. pages 79–85.

Ying Li, Zhenghua Li, and Min Zhang. 2020. Semi-supervised domain adaptation for dependency parsing via improved contextualized word representations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3806–3817, Barcelona, Spain (Online).

Zhenghua Li, Xue Peng, Min Zhang, Rui Wang, and Luo Si. 2019. Semi-supervised domain adaptation for dependency parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2386–2395, Florence, Italy.

Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2020. Finbert: A pre-trained financial language representation model for financial text mining. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4513–4519. International Joint Conferences on Artificial Intelligence Organization. Special Track on AI in FinTech.

Martin Malmsten, Love Börjeson, and Chris Haffenden. 2020. Playing with words at the national library of sweden – making a swedish bert.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330. Penn Treebank.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 62–72, Edinburgh, Scotland, UK.

Robert Östling. 2013. Stagger: an open-source part of speech tagger for Swedish. *Northern European Journal of Language Technology*, 3:1–18.

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

Slav Petrov and Dan Klein. 2007. Improved Inference for Unlexicalized Parsing. In *Proceedings of Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pages 404–411, Rochester, NY.

Slav Petrov and Ryan McDonlad. 2012. Overview of the 2012 shared task on Parsing the Web. In *Notes of the First Workshop on Syntactic Analysis of Non-Canonical Language (SANCL)*, Sapporo, Japan.

Barbara Plank and Gertjan van Noord. 2011. Effective measures of domain similarity for parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1566–1576, Portland, Oregon, USA.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in NLP—A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online).

Ines Rehbein and Felix Bildhauer. 2017. Data point selection for genre-aware parsing. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 95–105, Prague, Czech Republic.

Rudolf Rosa and Zdeněk Žabokrtský. 2015. KLcpos3 - a language similarity measure for delexicalized parser transfer. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 243–249, Beijing, China.

Kristján Rúnarsson and Einar Freyr Sigursson. 2020. Parsing Icelandic alingi transcripts: Parliamentary speeches as a genre. In *Proceedings of the Second ParlaCLARIN Workshop*, pages 44–50, Marseille, France. European Language Resources Association.

Sara Stymne, Miryam de Lhoneux, Aaron Smith, and Joakim Nivre. 2018. Parser training with heterogeneous treebanks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 619–625, Melbourne, Australia.

Heike Telljohann, Erhard Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2015. Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z). Seminar für Sprachwissenschaft, Universität Tübingen, Germany.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, Thomas R. McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019b. What do you learn from context? probing for sentence structure in contextualized word representations.

Clara Vania, Andreas Grivas, and Adam Lopez. 2018. What do character-level models learn about morphology? the case of dependency parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2573–2583, Brussels, Belgium.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: BERT for Finnish.

Wietse de Vries, Andreas van Cranenburgh, and Malvina Nissim. 2020. What's so special about BERT's layers? a closer look at the NLP pipeline in monolingual and multilingual models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4339–4350, Online. Association for Computational Linguistics.

Daniel Younger. 1967. Recognition and parsing of context-free languages in n^3. *Information and Control*, 10(2):189–208.

Yu Zhang, Houquan Zhou, and Zhenghua Li. 2020. Fast and accurate neural crf constituency parsing. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4046–4053.

Houquan Zhou, Yu Zhang, Zhenghua Li, and Min Zhang. 2020. Is pos tagging necessary or even helpful for neural dependency parsing? In *Natural Language Processing and Chinese Computing*, pages 179–191, Cham. Springer International Publishing.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.

A Full English Results for Parsers and BERT Models

Parser		Baseline	Answers	Email	Newsgroup	Reviews	Weblog	All	Gap
BNPno	Answers	88.72	NA	89.51	89.38	89.57	**89.67**	90.66 (+1.94)	+.99
	Email	89.29	**90.62**	NA	89.92	90.03	89.93	91.16 (+1.87)	+.54
	Newsgroup	87.61	**88.67**	88.34	NA	86.90	87.82	89.05 (+1.44)	+.38
	Reviews	88.68	**89.52**	89.22	89.46	NA	89.15	90.52 (+1.84)	+1.00
	Weblog	91.13	92.75	**92.83**	92.55	92.61	NA	93.75 (+2.62)	+.92
BNP	Answers	89.01	NA	89.63	89.43	**89.81**	89.37	90.66 (+1.65)	+.85
	Email	89.03	89.86	NA	89.94	**90.44**	89.35	91.21 (+2.18)	+.77
	Newsgroup	86.50	**87.46**	86.75	NA	86.75	87.54	88.20 (+1.70)	+.74
	Reviews	88.94	**89.84**	89.65	89.62	NA	89.60	90.20 (+1.26)	+.36
	Weblog	90.66	92.22	92.18	92.16	**92.47**	NA	93.53 (+2.87)	+1.06
SuPar	Answers	88.10	NA	89.16	89.23	**89.69**	89.44	90.54 (+2.44)	+.85
	Email	89.06	**90.62**	NA	90.27	89.98	90.09	91.34 (+2.28)	+.72
	Newsgroup	86.68	**88.13**	88.00	NA	87.47	87.78	89.42 (+2.74)	+1.29
	Reviews	88.06	89.35	89.19	89.08	NA	**89.57**	90.39 (+2.33)	+.82
	Weblog	90.58	91.95	91.94	**92.41**	92.05	NA	93.56 (+2.98)	+1.15

Table 3: Full EWT Results with mBERT on Dev Set

Parser		Baseline	Answers	Email	Newsgroup	Reviews	Weblog	All	Gap
BNPno	Answers	89.68	NA	90.45	90.51	**90.58**	90.41	91.48 (+1.80)	+.90
	Email	88.75	**90.67**	NA	90.13	90.31	90.56	91.35 (+2.60)	+.68
	Newsgroup	87.96	89.07	88.25	NA	88.95	**89.51**	90.29 (+2.33)	+.78
	Reviews	89.70	90.53	**90.65**	90.34	NA	90.06	90.76 (+1.06)	+.11
	Weblog	91.48	93.14	92.87	93.28	**93.39**	NA	95.01 (+3.53)	+1.62
BNP	Answers	89.60	NA	90.87	90.45	**90.93**	90.29	91.63 (+2.03)	+.70
	Email	89.28	90.10	NA	90.41	**90.58**	89.93	91.58 (+2.30)	+1.00
	Newsgroup	87.74	**89.01**	88.47	NA	88.69	88.73	89.85 (+2.11)	+.84
	Reviews	89.88	**90.83**	90.27	90.25	NA	90.02	90.69 (+.81)	-.14
	Weblog	91.28	93.23	93.17	92.69	**93.32**	NA	94.29 (+3.01)	+.97
SuPar	Answers	89.95	NA	90.98	90.90	90.93	**91.10**	92.19 (+2.24)	+1.09
	Email	90.14	**91.38**	NA	91.20	91.06	91.23	92.51 (+2.37)	+1.13
	Newsgroup	88.57	88.90	88.97	NA	89.55	**89.89**	90.91 (+2.34)	+1.02
	Reviews	89.28	90.17	90.39	89.76	NA	**90.69**	91.73 (+2.45)	+1.04
	Weblog	91.82	92.79	92.87	**93.17**	92.67	NA	94.31 (+2.49)	+1.14

Table 4: Full EWT Results with BERTbc on Dev Set

B Full Swedish Results for Parsers and BERT Models

Parser		Baseline	Blog	Europarl	Public	Novels	Wiki	All	Gap
BNPno	Blog	75.99	NA	76.65	77.09	78.01	**78.14**	79.22 (+3.23)	+1.08
	Europarl	80.91	82.60	NA	**82.97**	82.44	81.98	83.97 (+3.06)	+1.00
	Public	83.11	84.55	83.33	NA	**85.81**	84.34	85.34 (+2.23)	-.47
	Novels	85.14	**88.04**	87.03	87.67	NA	87.33	89.75 (+4.61)	+1.71
	Wiki	82.10	83.75	84.35	84.74	**84.89**	NA	84.68 (+2.58)	-.21
BNP	Blog	75.87	NA	77.45	77.86	**78.80**	78.18	79.57 (+3.70)	+.77
	Europarl	80.71	82.22	NA	83.20	**83.55**	83.06	84.41 (+3.70)	+.86
	Public	83.25	83.99	82.94	NA	**86.51**	84.77	86.72 (+3.47)	+.21
	Novels	84.94	**88.91**	86.99	86.81	NA	87.38	89.59 (+4.65)	+.68
	Wiki	81.84	83.56	83.42	**83.88**	83.56	NA	85.03 (+3.19)	+1.15
SuPar	Blog	74.40	NA	77.55	76.90	77.77	**79.18**	80.13 (+5.73)	+.95
	Europarl	82.52	82.60	NA	**84.36**	83.03	82.81	85.01 (+2.49)	+.65
	Public	84.95	85.98	85.06	NA	**86.88**	86.43	87.99 (+3.04)	+1.11
	Novels	86.54	88.58	88.67	**89.30**	NA	88.12	90.80 (+4.51)	+1.50
	Wiki	83.13	84.61	83.72	**85.57**	84.48	NA	86.12 (+2.99)	+.55

Table 5: Full SET Results with mBERT on Dev Set

Parser		Baseline	Blog	Europarl	Public	Novels	Wiki	All	Gap
BNPno	Blog	71.81	NA	73.82	75.93	74.62	**76.07**	77.87 (+6.06)	+1.80
	Europarl	78.21	78.73	NA	79.43	**80.78**	79.89	82.12 (+3.91)	+1.34
	Public	83.60	**84.16**	83.53	NA	81.99	83.09	86.30 (+2.70)	+2.14
	Novels	81.71	**85.13**	83.34	84.80	NA	85.08	87.86 (+6.15)	+2.73
	Wiki	78.52	80.32	**80.47**	80.40	80.46	NA	82.97 (+4.45)	+2.50
BNP	Blog	72.37	NA	74.69	**76.55**	76.19	75.79	79.12 (+6.75)	+3.46
	Europarl	78.56	80.70	NA	80.97	80.66	**81.85**	83.01 (+4.45)	+1.22
	Public	82.66	83.75	83.90	NA	83.34	**84.51**	86.98 (+4.32)	+2.47
	Novels	82.70	85.69	84.96	86.14	NA	**86.33**	88.44 (+5.74)	+2.11
	Wiki	78.45	80.04	79.92	**82.13**	81.26	NA	83.13 (+4.68)	+1.00
SuPar	Blog	73.11	NA	74.85	74.41	73.72	**75.86**	80.21 (+7.10)	+4.35
	Europarl	80.86	81.89	NA	**83.35**	81.95	82.20	84.32 (+3.46)	+.97
	Public	83.30	84.54	83.22	NA	**85.06**	84.41	86.50 (+3.20)	+1.44
	Novels	83.82	**86.22**	85.57	85.97	NA	85.50	89.04 (+5.22)	+2.82
	Wiki	80.31	82.18	82.18	**83.48**	81.77	NA	85.21 (+4.90)	+1.73

Table 6: Full SET Results with swBERT on Dev Set

Parser		Baseline	Blog	Europarl	Public	Novels	Wiki	All	Gap
BNPno	Blog	79.37	NA	81.67	81.70	82.63	**82.68**	84.25 (+4.88)	+1.57
	Europarl	84.38	86.60	NA	86.82	**87.34**	86.63	86.96 (+2.58)	+.14
	Public	89.03	89.66	89.85	NA	**90.47**	90.26	90.93 (+1.90)	+.46
	Novels	87.29	91.00	89.03	**91.39**	NA	90.53	92.74 (+5.45)	+1.35
	Wiki	84.21	86.23	86.66	86.41	**87.05**	NA	87.77 (+3.56)	+.72
BNP	Blog	78.63	NA	81.18	81.74	82.44	**82.48**	83.95 (+5.32)	+1.47
	Europarl	84.63	86.33	NA	**87.00**	86.42	86.36	87.75 (+3.12)	+.75
	Public	88.90	89.77	89.50	NA	**89.86**	89.00	91.35 (+2,45)	+1.49
	Novels	88.38	**91.78**	89.88	91.26	NA	91.39	92.41 (+4.03)	+.63
	Wiki	85.47	**86.94**	86.61	86.43	86.50	NA	87.61 (+2.14)	+.67
SuPar	Blog	79.36	NA	81.79	81.76	82.64	**83.28**	84.45 (+5.09)	+1.17
	Europarl	84.92	85.90	NA	86.54	86.54	**86.74**	88.18 (+3.26)	+1.44
	Public	87.53	88.46	88.34	NA	**90.00**	89.32	90.74 (+3.21)	+.74
	Novels	88.64	90.65	89.65	**91.67**	NA	90.93	93.04 (+4.40)	+1.37
	Wiki	85.96	87.70	86.88	**87.93**	87.37	NA	89.26 (+3.30)	+1.33

Table 7: Full SET Results with kbBERT on Dev Set