Probing the SpanBERT Architecture to interpret Scientific Domain Adaptation Challenges for Coreference Resolution

Hari Timmapathini, Anmol Nayak, Sarathchandra Mandadi, Siva Sangada, Vaibhav Kesri, Karthikeyan Ponnalagu, Vijendran Venkoparao

ARiSE Labs at Bosch

{HariPrasad.Timmapathini, Anmol.Nayak, Mandadi.Sarathchandra, SivaChaitanya.Sangada, Vaibhav.Kesari, Karthikeyan.Ponnalagu, GopalanVijendran.Venkoparao}@in.bosch.com

Abstract

Coreference Resolution is a challenging problem in Natural Language Processing (NLP) that aims at clustering all references of the same entity or event. This requires both syntactic and semantic understanding of the text. A strong coreference resolution model is essential for achieving good performance in several downstream NLP tasks such as Question-Answering, Information Extraction etc. SpanBERT (Joshi et al. 2020) has achieved state of the art performance in coreference resolution on the OntoNotes dataset (Pradhan et al. 2012). However it still has several challenges when performing coreference resolution on documents involving multiple domain specific entities and events. In this paper we have highlighted these issues with SpanBERT-Base (pre-trained coreference model) in scientific domain adaptation. Our detailed experiments have been performed on the SciERC scientific abstract dataset (Luan et al. 2018), where we analyse the encoder attention and probe the coarse-to-fine head network to interpret the short comings of SpanBERT. This has lead to interesting findings that showed: 1) While we observed that the syntactic behaviour is captured appropriately, the self-attention mechanism in the encoder layers of SpanBERT struggles to capture domain specific semantic concepts, 2) Inferior mention spans are picked in the top mention spans list due to poor mention scores even though better candidate key mention spans exist, and 3) Even by increasing the hyperparameter λ from 0.4 to 1 and 2, there is insignificant improvement in both $N_{key \cap response}$ and response coreference cluster scores across 5 different evaluation metrics.

Introduction

BERT (Devlin et al. 2019) has been a breakthrough in language understanding by leveraging the multi-head self-attention mechanism (Vaswani et al. 2017) in its architecture. It is one of the prominent models used for a variety of NLP tasks. With the Masked Language Model (MLM) method, it has been successful at leveraging bidirectionality while training the language model. SpanBERT-Base model has 12 encoder layers, with each layer consisting of 12 self-attention heads. The word representations are context-dependent 768 dimensional dynamic embeddings. The vocabulary size is 28996 and contains 101 unused slots. The unused slots in the vocabulary can be used to include domain specific words, however the representations of these will have to be fine-tuned with domain specific corpus.

While the BERT architecture relies on MLM at word level and Next Sentence Prediction (NSP) during training, SpanBERT has changed the learning mechanism to MLM at span level and uses a Span Boundary Objective (SBO). SBO predicts a target masked token by using the representations of the boundary tokens of a given span along with the positional embedding of the target masked token. This learning mechanism has enabled SpanBERT to outperform BERT on almost all tasks with significant improvements. For the coreference resolution task, SpanBERT leverages an independent implementation of higher order coarse-to-fine span ranking architecture (Lee, He, and Zettlemoyer 2018) that iteratively refines the mentions using an attention mechanism.

A strong coreference resolution model is essential in domains which describe concepts that require long range dependencies between mentions for applications like Question-Answering systems, Information Extraction for Domain Specific Knowledge Graphs (Lin et al. 2017; Kejriwal 2019). Scientific domain adaptation within industries is challenging due to the following reasons:

1. Typically there is a lack of sufficient data to fine-tune the language model of such large pre-trained networks.
2. Unavailability of annotated data for task specific fine-tuning, as it requires a domain expert's understanding to annotate the data correctly to encapsulate the nuances of the domain.

We probe the model to analyse 5 different aspects of the SpanBERT coreference resolution architecture: Encoder attention, Identification of Mentions, Mention scores, Antecedent scores and Coreference Clusters. The *Newswire* genre of OntoNotes was selected with SpanBERT. MUC, B^3, $CEAF_m$, $CEAF_e$ and LEA (Pradhan et al. 2014; Moosavi and Strube 2016) have been selected as the coreference evaluation metrics. The experiments are performed on the SciERC dataset along with motivating example sentences, that depict the various kinds of sentence structures typically found in technical documents of AUTOSAR (http://www.autosar.org/) compliant automotive domain systems. We discuss these challenges below by analysing SpanBERT Encoder and Probing the Coarse-to-fine network.

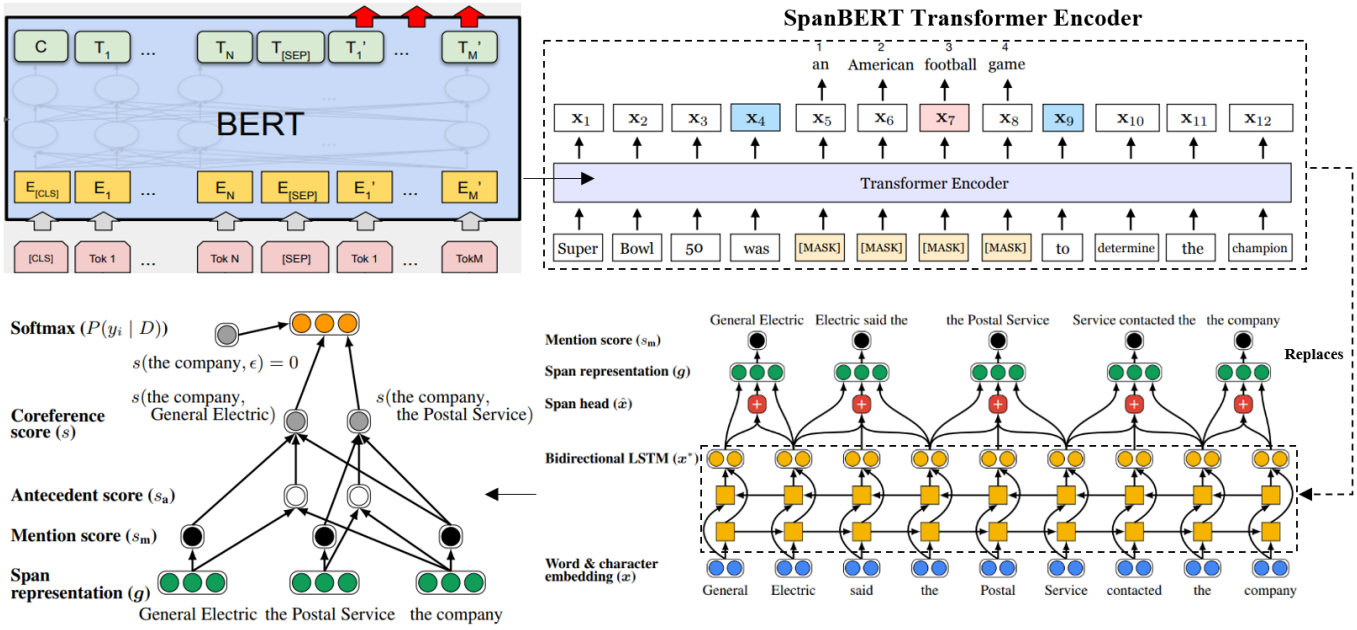

Figure 1: SpanBERT Coreference Resolution Architecture. Sources: (Devlin et al. 2019), (Joshi et al. 2020), (Lee et al. 2017)

Background

SpanBERT Coreference Resolution architecture consists of a SpanBERT Transformer Encoder with a Coarse-to-fine head network (Figure 1). The input is tokenized with a BERT variant of the WordPiece algorithm (Schuster and Nakajima 2012) and passed into the encoder to generate contextualized representations for each token. Mention spans are non-overlapping segments from the input text upto a predefined length. The encoder representations are consumed by the coarse-to-fine network and iteratively refined using an attention mechanism to give the span representations *g* which are used for computing the following Coreference resolution specific scores:

1. Mention score $s_m(i)$ for a mention span *i*, that is used to further prune the mention spans list.
2. Fast antecedent score $s_c(i, j)$ between mention span *i* and candidate antecedent span *j*, that uses a bi-linear scoring function to pick the top K candidate antecedent spans for each mention.
3. Antecedent distance score $s_d(i, j)$ that is computed using 10 semi-log scale buckets.
4. Slow antecedent score $s_a(i, j)$ that relies upon mention span *i* and candidate antecedent span *j* representations, element-wise similarity between *i* and *j*, and a feature vector encoding genre information, span distance etc.
5. Coreference resolution score $s(i, j)$ that is used to decide whether candidate antecedent span *j* is coreferent to mention span *i*.

Further, the mention spans can be segregated into 3 categories:

- Key spans M_{key}, which are the annotated gold standard spans.
- Top spans M_{top}, which are the final pruned set of candidate mention spans selected by the coarse-to-fine network.
- Response spans $M_{response}$, which are the system generated output spans found in the predicted coreference clusters. These are a subset of the Top spans.

We evaluated the overall coreference resolution performance of SpanBERT using 5 standard metrics, each of which compute the Precision, Recall and F1 scores with emphasis on different aspects of the coreference clusters (Cai and Strube 2010):

- MUC: It is a link-based metric that computes the minimum number of links between mentions to be inserted or deleted when mapping a system generated response to a gold standard key set.
- B^3: It is a mention-based metric that computes the overall Precision and Recall based on the Precision and Recall of the individual mentions.
- $CEAF_m$: It is a mention-based variant of the CEAF metric, which indicates the percentage of mentions that are in the correct entities.
- $CEAF_e$: It is an entity-based variant of the CEAF metric, which indicates the percentage of correctly recognized entities.
- LEA: It is a link-based entity-aware metric that considers how important the entity is and how well it is resolved.

We also performed a baseline comparison between the independent variants of SpanBERT-Base and BERT-Base (Joshi et al. 2019) pretrained coreference models on the SciERC dataset.

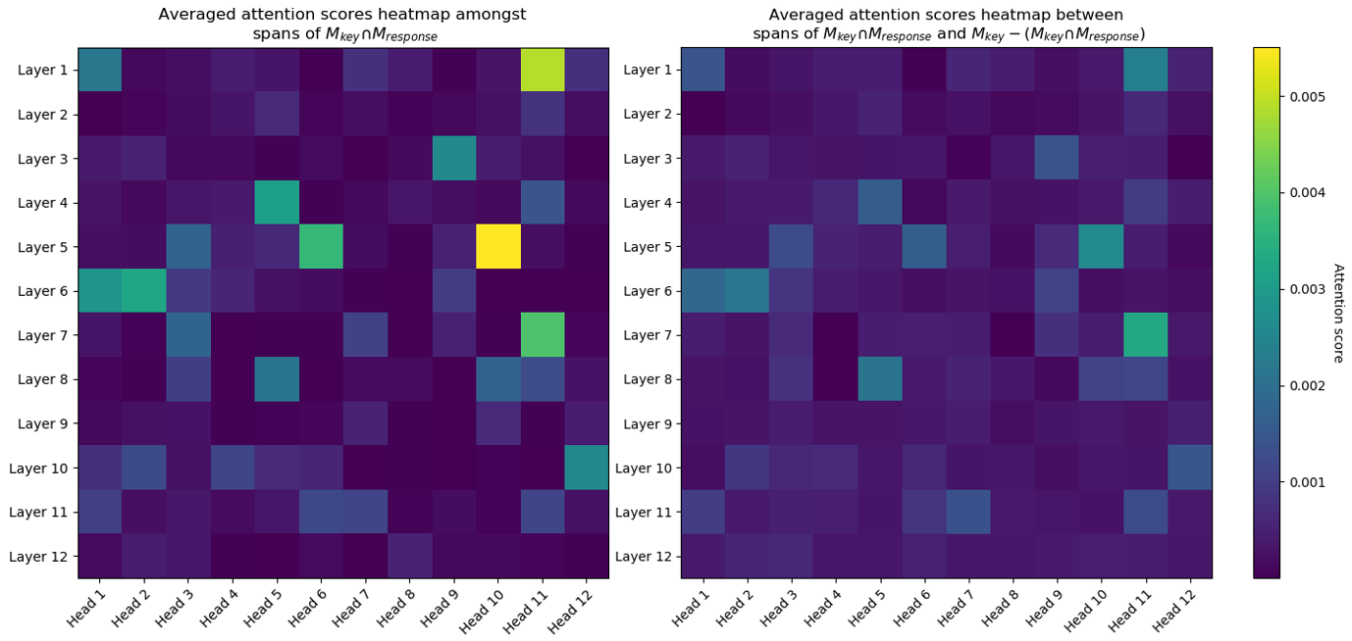

Figure 2: Attention scores heatmaps for SpanBERT encoder layers on the SciERC dataset.

Analysis of SpanBERT Encoder

BERT has been shown to learn surface level features in the early layers, syntactic features in the middle layers and semantic features in the higher layers (Jawahar, Sagot, and Seddah 2019). Coreference resolution relies heavily on capturing the syntactic behaviour to pick syntactically plausible mention spans. BERT has been previously shown to capture strong syntactic representations (Tenney et al. 2019).

We found that across the SciERC scientific abstracts, most of the top spans selected by SpanBERT had the correct boundaries. This strong syntactic understanding in SpanBERT can be attributed to the SBO technique it utilizes during training. While the SpanBERT training objectives have improved the span boundaries, domain specific semantic concepts are significantly more difficult to learn due to the following reasons:

1. Events typically involve multiple entities interacting under certain conditions.
2. Long range dependencies between coreferent mentions as sentences tend to build upon concepts previously mentioned.

To see how SpanBERT handles this, we analyse the self-attention in the encoder layers between two sets of mention spans for each abstract in the SciERC dataset:

- Set 1: Pairwise attention scores amongst spans in $M_{key} \cap M_{response}$.
- Set 2: Pairwise attention scores between spans in $M_{key} \cap M_{response}$ and M_{key} - ($M_{key} \cap M_{response}$).

A sample output for the different categories of mention spans and clusters for an abstract from the SciERC coreference resolution dataset can be seen in Table 1. For each encoder layer, we extract the pairwise attention scores to observe the difference in attention given by a clustered key span to a co-occurring clustered key span in comparison to a non-clustered key span. Across the 12 layers we observed that the attention scores in Set 1 and Set 2 were extremely small. While we observed that the dominant heads (shades of yellow and green in Figure 2) in both Set 1 and Set 2 tend to be the same, on average each pairwise attention score for these heads was found to be less than 0.01, which is less than 1% of the total attention mass for the abstract. As the attention scores are computed from the Key and Query vectors of a given word, these extremely low attention scores reflect the weak semantic representations of the spans.

Further, this also highlights that no specific head across the 12 encoder layers is exhibiting strong coreference behaviour in the case of scientific domain abstracts. Previously, BERT showed that the different heads of each layer attend to specific linguistic behaviours like coreference, syntax, delimiter tokens (Clark et al. 2019). This semantic loss leads to cascading problems in the coarse-to-fine network due to the Fast and Slow antecedent scores computation. The weak semantic representations have also lead to lesser number of key mention spans being picked up as candidates to be clustered (Table 2). This shows that the self-attention mechanism in the encoder layers of SpanBERT struggles to capture scientific domain specific semantic concepts.

Probing the Coarse-to-fine network

SpanBERT uses a coarse-to-fine architecture in the head network to perform coreference resolution. For a given sentence, the network first generates the mention scores for all possible candidate mentions. It then picks the top *M=min(3900,λT)* non-crossing mentions based on the men-

Abstract ID	C90-3007
Abstract Text	This paper examines the properties of feature-based partial descriptions built on top of Halliday's systemic networks. We show that the crucial operation of consistency checking for such descriptions is NP-complete, and therefore probably intractable, but proceed to develop algorithms which can sometimes alleviate the unpleasant consequences of this intractability.
$\mathbf{M}_{key}$	[feature-based partial descriptions; descriptions]
$\mathbf{M}_{top}$	[This paper; feature-based partial descriptions built on top of Halliday's systemic networks; such descriptions; intractable; this intractability; ...]
$\mathbf{M}_{response}$	[intractable; this intractability]
$\mathbf{M}_{key \cap top}$	[]
$\mathbf{M}_{key \cap response}$	[]
Key clusters	[feature-based partial descriptions; descriptions]
Response clusters	[intractable; this intractability]

Table 1: Sample output mention spans and clusters with SpanBERT-Base (λ = 0.4) for abstract ID C90-3007 of the SciERC coreference resolution dataset.

λ	$\mathbf{N}_{key}$	$\mathbf{N}_{top}$	$\mathbf{N}_{response}$	$\mathbf{N}_{key \cap top}$	$\mathbf{N}_{key \cap response}$	$\mathbf{P}_{response}$	$\mathbf{R}_{response}$	$\mathbf{F1}_{response}$
0.4	2686	32102/32729	3750/3136	788/888	381/356	10.16/11.35	14.18/13.25	11.84/12.23
1	2686	74509/74713	3730/3113	1076/1248	383/356	10.26/11.43	14.26/13.25	11.94/12.28
2	2686	126395/126332	3686/2994	2222/2295	381/353	10.33/11.79	14.18/13.14	11.96/12.43

Table 2: Identification of Mentions metrics with SpanBERT-Base/BERT-Base (λ = 0.4, 1, 2) on the SciERC coreference resolution dataset.

	P			**R**			**F1**		
Metric	**λ=0.4**	**λ=1**	**λ=2**	**λ=0.4**	**λ=1**	**λ=2**	**λ=0.4**	**λ=1**	**λ=2**
MUC	5.02	5.1	5.16	7	7.06	7.06	5.85	5.92	5.96
B^3	6.16	6.3	6.34	7.89	7.94	7.9	6.92	7.02	7.04
$CEAF_m$	9.8	9.91	9.97	13.7	13.77	13.7	11.43	11.53	11.54
$CEAF_e$	8.83	8.97	9.01	12.4	12.54	12.45	10.32	10.46	10.46
LEA	3.78	3.89	3.93	4.7	4.74	4.73	4.19	4.27	4.29
Avg.	**6.718**	**6.834**	**6.882**	**9.138**	**9.21**	**9.168**	**7.742**	**7.84**	**7.858**

Table 3: Coreference Resolution metrics with SpanBERT-Base (λ = 0.4, 1, 2) on the SciERC coreference resolution dataset.

	P			**R**			**F1**		
Metric	**λ=0.4**	**λ=1**	**λ=2**	**λ=0.4**	**λ=1**	**λ=2**	**λ=0.4**	**λ=1**	**λ=2**
MUC	5.54	5.59	5.69	6.22	6.22	6.1	5.86	5.89	5.89
B^3	6.66	6.75	6.93	7.12	7.09	7.02	6.89	6.92	6.98
$CEAF_m$	10.73	10.87	11.13	12.54	12.62	12.43	11.56	11.68	11.75
$CEAF_e$	9.08	9.16	9.44	11.31	11.38	11.24	10.07	10.15	10.26
LEA	3.9	3.97	4.04	4.06	3.98	3.96	3.98	3.98	4
Avg.	**7.182**	**7.268**	**7.446**	**8.25**	**8.258**	**8.15**	**7.672**	**7.724**	**7.776**

Table 4: Coreference Resolution metrics with BERT-Base (λ = 0.4, 1, 2) on the SciERC coreference resolution dataset.

tion scores, where *T* is the number of words in the tokenized sentence, and λ is a configurable parameter that decides the number of spans per word and is set to 0.4 (default) in SpanBERT coreference resolution.

We conducted our experiments with λ = 0.4, 1 and 2 to make sure that the limited size of the top span list is not a reason for key mentions to be discarded. It should be noted that while λ = 1 and λ = 2 may increase the number of key mention spans in the top span list, it comes at a performance cost as it can be seen in Table 2, N_{top} (λ = 2) $\approx 4 \times N_{top}$ (λ = 0.4).

For each of the top *M* mentions, top *K=min(50,λT)* antecedents are picked from the top mention span list based on the score $s_m(i)$+$s_m(j)$+$s_c(i,j)$+$s_d(i,j)$, where $s_m(i)$ is the mention score of mention span *i*, $s_m(j)$ is the mention score of antecedent span *j*, $s_c(i,j)$ is the fast antecedent score between spans *i* and *j*, and $s_d(i,j)$ is the antecedent distance score introduced in the coarse-to-fine implementation of SpanBERT. From this pruned set of antecedents, final coreference score $s(i,j)$=$s_m(i)$+$s_m(j)$+$s_c(i,j)$+$s_d(i,j)$+$s_a(i,j)$ is calculated between each pair of mention and its top an-

Sl. No.	Sentences
1.	When cruise control button is pressed for 2 seconds ***cruise control is activated***$_1$. After ***this***$_2$ happens, the speed is maintained.
2.	After ***this condition***$_3$ is satisfied, cruise control will be activated: ***Cruise control button is pressed for 2 seconds***$_4$.
3.	When the ***cruise control button is pressed for 2 seconds***$_5$, ***then***$_6$ cruise control is activated.
4.	***Adaptive Cruise control***$_7$, commonly known as ***Cruise control***$_8$, is a speed maintaining feature that is often found in high-end cars.
5.	***Cruise control***$_9$ is a speed maintain feature. When the car is ***cruising***$_{10}$, a beep is triggered every 5 minutes.
6.	When the ***minimum speed threshold***$_{11}$ of ***Cruise control***$_{12}$ is reached, the cruise activation lamp turns green to signify cruise control activation is available.
7.	Cruise control is usually available in ***high-end cars***$_{13}$. ***Such vehicles***$_{14}$ are typically 30% costlier than mid-end cars.
8.	When the ***vehicle speed***$_{15}$ is above ***60kmph***$_{16}$, cruise control is activated.

Table 5: Automotive domain motivating example sentences (Sentence-wise coreference clusters in bold; Span ID in subscript).

SpanBERT Clusters	Mention ID$_i$	Antecedent ID$_j$	$s_m(i)$	$s_m(j)$	$s_c(i,j)$	$s_a(i,j)$	$s_d(i,j)$	$s(i,j)$
(this,activated)	2	1	-15.409	**-30.980**	-	-	-	-
None found	4	3	-25.474	-5.310	-25.678	-40.415	0.221	-96.656
None found	6	5	**-29.705**	-54.061	-	-	-	-
None found	8	7	-17.818	-27.222	-5.451	-8.502	0.214	-58.779
None found	10	9	**-28.193**	-11.187	-	-	-	-
None found	12	11	-7.134	**-56.156**	-	-	-	-
None found	14	13	-18.432	-9.841	40.200	-15.273	-0.159	-3.505
None found	16	15	**-32.280**	**-42.714**	-	-	-	-

Table 6: Clusters and Coarse-to-fine scores for the motivating example sentences (Spans not picked as top span by SpanBERT with λ = 0.4 are indicated by scores in bold).

tecedents, where $s_a(i,j)$ is the slow antecedent score. The top scoring antecedent *j* is then picked as a coreferent to the mention *i* if $s(i,j)>0$. Antecedents that result in a positive coreference score are only picked since a dummy antecedent is introduced before the softmax layer, whose coreference score with every mention is 0.

SpanBERT performance on the SciERC dataset

The SciERC dataset consists of 500 annotated scientific domain abstracts. The total number of key mention spans was 2686. We probed the coarse-to-fine head network to analyse two aspects of the SpanBERT coreference resolution architecture:

1. Qualitative and Quantitative measures of the Mention Spans (Table 2): Picking the top mention spans is the first important task for the head network. We observed that for λ = 0.4 and λ = 1, the recall of key mention spans is around 30% and 40% respectively. The recall increased to around 82% in the case of λ = 2. However that was only possible because 126395 top spans had to be picked, which is extremely large. The precision of the top spans was found to be extremely low for all the values of λ.

   We then checked the number of key mention spans that were part of the response clusters ($N_{key \cap response}$). In this case, for all the the values of λ the numbers turned out to be roughly the same. This clearly indicated that while increasing the value of λ increases the chances of a larger number of key mention spans to be part of the top spans list, it does not guarantee improvement in the number of key mentions becoming part of the response clusters.

   Across all the values of λ the Precision, Recall and F1 scores for the identification of mentions were found to be roughly 10%, 14% and 11% respectively. We believe that these low values are due to the weak SpanBERT representations for the mention spans found in the scientific domain abstracts which makes it difficult for the coarse-to-fine head network to recover from.

2. Overall coreference resolution performance (Table 3): We evaluated the SpanBERT coreference resolution performance using 5 different metrics, each of which target different aspects of the coreference clusters. Another indication that increasing the λ did not have significant improvement to the coreference resolution was that the Precision, Recall and F1 scores for coreference resolution were roughly the same being around 6%, 9% and 7% respectively.

The low scores appearing consistently both in Identification of Mentions and Overall coreference resolution across a large number of abstracts clearly indicates the difficulty that SpanBERT faces while adapting to the scientific domain corpus coreference resolution task. We also observed a similar performance in both Identification of Mentions (Table 2) and Overall coreference resolution (Table 4) with BERT-Base.

SpanBERT performance on the Automotive domain motivating example sentences

To get more granular insights into the coarse-to-fine network, we further probed the head network on the automotive domain motivating example sentences (Table 5) to extract the Mention scores, Fast Antecedent scores, Slow Antecedent scores, Antecedent distance scores and Final Coreference scores. SpanBERT did not give a valid coreference cluster for any of motivating example sentences (Table 6). In the first motivating example sentence, a cluster was found between *this* and *activated*, however it was still not the expected cluster. For the mentions which were not picked as top spans, $s_c(i,j)$, $s_a(i,j)$, $s_d(i,j)$ and $s(i,j)$ scores cannot be computed. We observed that:

- Due to the limit on the number of top mentions that can be picked, many expected mentions were eliminated due to a lower mention score. This happened in 5 different motivating example sentences, each of which had a different sentence structure.
- Even by increasing the λ to $\lambda = 1$ and $\lambda = 2$, the expected antecedents were eliminated from being part of the top span list by another irrelevant crossing mention that had a better mention score.

  For e.g. in the first motivating example sentence, the expected antecedent span *cruise control is activated* with a mention score of -30.980 was not picked as a top span, since a better scoring but irrelevant crossing mention *is activated . After this happens* received a mention score of -29.048.
- These different scores provide insights into the reasons behind certain clusters not being formed by the network.

We believe that probing the coarse-to-fine network reveals the underlying issue of the mention spans having weak semantic representations. Stronger semantic representations would lead to better mention scores for the expected mention spans, thereby ranking them higher to be selected as a top mention. This would also positively impact the antecedent scores as they rely heavily upon the mention and antecedent representations.

Conclusion and Future Work

We presented an analysis on the challenges faced by SpanBERT Coreference Resolution in tackling scientific domain corpus. We performed detailed experiments analysing the attention mechanism in the SpanBERT encoder layers along with probing the coarse-to-fine head network to understand how well the syntactic and semantic behaviours are being captured. Our findings show that while SpanBERT has a strong syntactic understanding, its semantic understanding of scientific domain documents is weak which further leads to cascading problems for the coreference resolution task. We believe that some of the directions which could improve the scientific domain adaptation of SpanBERT are:

1. As SpanBERT relies on the BERT variant of the WordPiece algorithm to tokenize an input text, which has previously been shown to give poorer performance in the case of Out-of-Vocabulary (OOV) words (Nayak et al. 2020), a frequency or likelihood based tokenization algorithm such as BPE-Dropout (Provilkov, Emelianenko, and Voita 2019), SentencePiece (Kudo and Richardson 2018) could lead to better sub-word choices and thereby better semantic representations for OOV words.
2. In the case where sufficient data exists to fine-tune the language model of SpanBERT, care should be taken to ensure that task specific catastrophic forgetting is avoided by leveraging advanced fine-tuning techniques (Dodge et al. 2020; Howard and Ruder 2018).

References

Cai, J.; and Strube, M. 2010. Evaluation metrics for end-to-end coreference resolution systems. In *Proceedings of the SIGDIAL 2010 Conference*, 28–36.

Clark, K.; Khandelwal, U.; Levy, O.; and Manning, C. D. 2019. What Does BERT Look At? An Analysis of BERT's Attention. In *BlackBoxNLP@ACL*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 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, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics. doi:10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

Dodge, J.; Ilharco, G.; Schwartz, R.; Farhadi, A.; Hajishirzi, H.; and Smith, N. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305* .

Howard, J.; and Ruder, S. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146* .

Jawahar, G.; Sagot, B.; and Seddah, D. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3651–3657. Florence, Italy: Association for Computational Linguistics. doi:10.18653/v1/P19-1356. URL https://www.aclweb.org/anthology/P19-1356.

Joshi, M.; Chen, D.; Liu, Y.; Weld, D. S.; Zettlemoyer, L.; and Levy, O. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics* 8: 64–77.

Joshi, M.; Levy, O.; Weld, D. S.; and Zettlemoyer, L. 2019. BERT for Coreference Resolution: Baselines and Analysis. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Kejriwal, M. 2019. *Domain-Specific Knowledge Graph Construction*. Springer.

Kudo, T.; and Richardson, J. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proceedings of the*

2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 66–71. Brussels, Belgium: Association for Computational Linguistics. doi:10.18653/v1/D18-2012. URL https://www.aclweb.org/anthology/D18-2012.

Lee, K.; He, L.; Lewis, M.; and Zettlemoyer, L. 2017. End-to-end Neural Coreference Resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 188–197. Copenhagen, Denmark: Association for Computational Linguistics. doi:10.18653/v1/D17-1018. URL https://www.aclweb.org/anthology/D17-1018.

Lee, K.; He, L.; and Zettlemoyer, L. 2018. Higher-Order Coreference Resolution with Coarse-to-Fine Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 687–692. New Orleans, Louisiana: Association for Computational Linguistics. doi:10.18653/v1/N18-2108. URL https://www.aclweb.org/anthology/N18-2108.

Lin, Z.-Q.; Bing, X.; Yan-Zhen, Z.; Jun-Feng, Z.; Xuan-Don, L.; Jun, W.; Hai-Long, S.; and Gang, Y. 2017. Intelligent development environment and software knowledge graph. *Journal of Computer Science and Technology* 242–249.

Luan, Y.; He, L.; Ostendorf, M.; and Hajishirzi, H. 2018. Multi-Task Identification of Entities, Relations, and Coreferencefor Scientific Knowledge Graph Construction. In *Proc. Conf. Empirical Methods Natural Language Process. (EMNLP)*.

Moosavi, N. S.; and Strube, M. 2016. Which Coreference Evaluation Metric Do You Trust? A Proposal for a Link-based Entity Aware Metric. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 632–642. Berlin, Germany: Association for Computational Linguistics. doi:10.18653/v1/P16-1060. URL https://www.aclweb.org/anthology/P16-1060.

Nayak, A.; Timmapathini, H.; Ponnalagu, K.; and Venkoparao, V. G. 2020. Domain adaptation challenges of BERT in tokenization and sub-word representations of Out-of-Vocabulary words. In *Proceedings of the First Workshop on Insights from Negative Results in NLP*, 1–5.

Pradhan, S.; Luo, X.; Recasens, M.; Hovy, E.; Ng, V.; and Strube, M. 2014. Scoring Coreference Partitions of Predicted Mentions: A Reference Implementation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 30–35. Baltimore, Maryland: Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P14-2006.

Pradhan, S.; Moschitti, A.; Xue, N.; Uryupina, O.; and Zhang, Y. 2012. CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, 1–40. Jeju Island, Korea: Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W12-4501.

Provilkov, I.; Emelianenko, D.; and Voita, E. 2019. BPE-Dropout: Simple and Effective Subword Regularization. *arXiv preprint* arXiv:1910.13267. URL https://arxiv.org/abs/1910.13267.

Schuster, M.; and Nakajima, K. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5149–5152. IEEE.

Tenney, I.; Xia, P.; Chen, B.; Wang, A.; Poliak, A.; McCoy, R. T.; Kim, N.; Van Durme, B.; Bowman, S. R.; Das, D.; et al. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. *arXiv preprint arXiv:1905.06316* .

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.