Pretraining with Contrastive Sentence Objectives Improves Discourse Performance of Language Models

Dan Iter[**], **Kelvin Guu**[**], **Larry Lansing**[**], and **Dan Jurafsky**[*]

[*]Computer Science Department, Stanford University
[**]Google Research
{daniter,jurafsky}@stanford.edu
{kguu,llansing}@google.com

Abstract

Recent models for unsupervised representation learning of text have employed a number of techniques to improve contextual word representations but have put little focus on discourse-level representations. We propose CONPONO[1], an inter-sentence objective for pretraining language models that models discourse coherence and the distance between sentences. Given an anchor sentence, our model is trained to predict the text k sentences away using a sampled-softmax objective where the candidates consist of neighboring sentences and sentences randomly sampled from the corpus. On the discourse representation benchmark DiscoEval, our model improves over the previous state-of-the-art by up to 13% and on average 4% absolute across 7 tasks. Our model is the same size as BERT-Base, but outperforms the much larger BERT-Large model and other more recent approaches that incorporate discourse. We also show that CONPONO yields gains of 2%-6% absolute even for tasks that do not explicitly evaluate discourse: textual entailment (RTE), common sense reasoning (COPA) and reading comprehension (ReCoRD).

1 Introduction

Pretraining large language models has become the primary method for learning representations from unsupervised text corpora. Since the initial improvements demonstrated by ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019), many alternative pretraining methods have been proposed to best leverage unlabeled data. These methods include bi-directional language modeling (Peters et al., 2018), masked language models (Devlin et al., 2019), word order permutation (Yang et al., 2019), more robust training (Liu et al., 2019) and more efficient architectures (Lan et al., 2019). However, little focus has been put on learning discourse coherence as part of the pretraining objective.

While discourse coherence has been of great interest in recent natural language processing literature (Chen et al., 2019; Nie et al., 2019; Xu et al., 2019), its benefits have been questioned for pretrained language models, some even opting to remove any sentence ordering objective (Liu et al., 2019). However, in a recently published benchmark for evaluating discourse representations, Chen et al. (2019) found that the best performing model was surprisingly BERT, despite comparing against models specifically designed for discourse, such as DisSent (Nie et al., 2019) and a new recurrent network trained on a large range of sentence ordering objectives. We show that combining transformer encoders with our inter-sentence coherence objective, we can further improve discourse-level representations in language models.

We present a model that trains a sentence-level encoder to capture discourse relationships between sentences, including ordering, distance and coherence. The encoder is trained by using its output to predict spans of text that are some k sentences away from a context in either direction. The predictions are made discriminatively with a sampled-softmax that contrasts the correct target sentence against negatives, including hard examples sampled from the same paragraph. Our objective is inspired by the recently proposed Constrastive Predictive Coding (CPC) (van den Oord et al., 2018), but, among other differences, is applied on the sentence-level rather than the token-level and is bi-directional. We call this the CONtrastive Position and Ordering with Negatives Objective (CONPONO)[2].

[*] Work done during internship at Google.

[1] Code is available at https://github.com/google-research/language/tree/master/language/conpono and https://github.com/daniter-cu/DiscoEval

[2] Also means *arrange* or *order* in Latin.

We evaluate our model on DiscoEval (Chen et al., 2019), a recently published benchmark for evaluating and probing for various aspects of discourse-level semantics in representations output by discourse models. We observe that the representations learned with CONPONO outperform BERT-Large and achieve a new state-of-the-art despite using fewer parameters and training on the same data. Furthermore, we show that our new objective improves model performance on other tasks including textual entailment, common-sense reasoning and reading comprehension. We compare CONPONO against BERT-Base on RTE (Giampiccolo et al., 2007; Bentivogli et al., 2009), COPA (Roemmele et al., 2011) and ReCoRD (Zhang et al., 2018), while controlling for model size, training data and training time.

Our main contributions are:

1. We describe a novel sentence-level discourse objective that is used in conjunction with a masked language model for unsupervised representation learning for text. We show that this objective can leverage the cross-attention and pretrained weights of a transformer model to learn discourse-level representations.

2. We show that our model achieves a new state-of-the-art on DiscoEval, improving the results on 5 of the 7 tasks and increasing accuracy by up to 13% and an average of over 4% absolute across all tasks. We also show 2%-6% absolute improvements over Bert-Base on RTE, COPA and ReCoRD as evidence that discourse pretraining can also improve model performance on textual entailment, common-sense reasoning and reading comprehension.

2 Model

Figure 1 illustrates the CONPONO model. The intuition is that if the model is able to accurately predict the surrounding target sentences given some anchor text, then the vector representations for these sentences should also be useful for downstream tasks.

The input to the model is a paragraph that is split into sentences. A sentence is chosen at random as the anchor, and will be denoted as s_i. We encode s_i with a transformer encoder to produce a vector c_i. The surrounding sentences are denoted as s_{i+k} where $k \in [-K \mathinner{..} -1, 1 \mathinner{..} K]$, meaning the maximum distance we use is K. We report results for $K \in [1..4]$. These sentences, s_{i+k}, are encoded jointly with the anchor sentence. We use just a single encoder g_θ so all text is encoded with the same weights. The encoded vectors are named t_{i+k} because these are the target vectors the model tries to identify given the anchor and a target distance k. Equation 1 defines t_{i+k} and c_i as a function g_θ of the input sentences. Note that the CONPONO g_θ is different from the encoder in CPC because we input both the anchor and the target into the encoder, rather than separate anchor and target encoders.

$$t_{i+k} = g_\theta(s_i, s_{i+k}), \quad c_i = g_\theta(s_i) \tag{1}$$

Given the anchor and targets, we define a log-bilinear model in equation 2 to score the plausibility of target t_{i+k} being in position k from anchor c_i. The full set of parameters for our model is θ for the encoder and a W_k for each k. CPC has the same bi-linear form as Equation 2 but the architecture for the encoders is different.

$$f_k(s_{i+k}, s_i) = \exp(t_{i+k}^T W_k c_i) \tag{2}$$

The loss for each k is given in equation 3 where the score for the correct target is contrasted to scores of random samples s_j, sampled from both in-document and random sentences from the corpus, S.

$$\mathcal{L}_k = -\mathbb{E}_S \left[log \frac{f_k(s_{i+k}, s_i)}{\Sigma_{s_j \in S} f_k(s_j, s_i)} \right] \tag{3}$$

To train CONPONO, we sample negative examples randomly from the corpus and from the same paragraph but different k as hard negatives. Note that when $|k|$ is greater than 1, there will be sentences between the anchor sentence and target sentence that will be purposely omitted from the input. The missing context is intended to create a challenging objective where the model may not be able to rely on trivial signals that often appear in contiguous sentences.

2.1 Encoder Architectures

For each example we encode two text spans, the anchor and the target. There are three main options for encoding the two spans into c_i and t_{i+k}. The simplest method, and most similar to CPC is to encode the anchor and target separately, which we call *isolated encoding*. With this encoder, equation 1 will be $t_{i+k} = g_\theta(s_{i+k})$. The major drawback of this approach is that there is no token-level cross-attention between the anchor and the target, which has been shown to generally improve text encoding

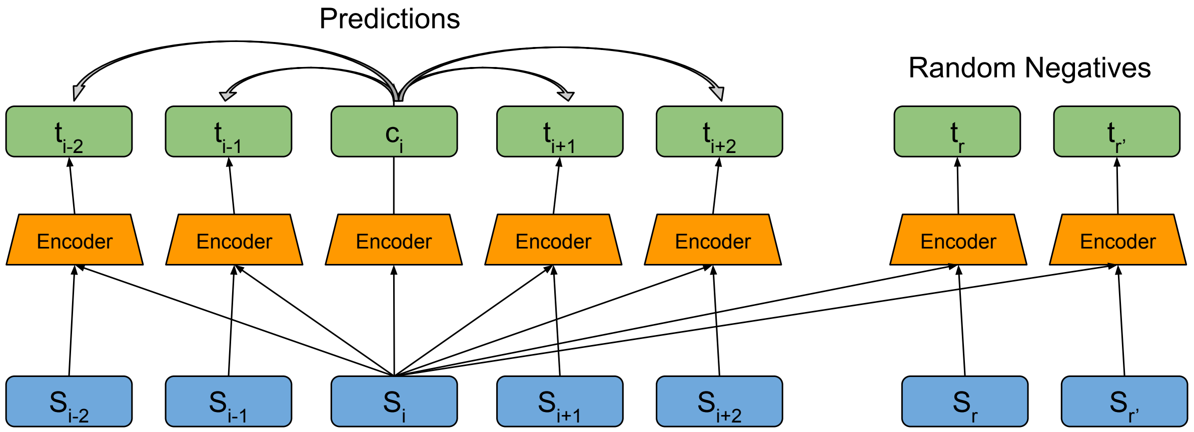

Figure 1: During training, a text segment is selected as the anchor (S_i). The anchor as well as all the targets, $S_{i-2}...S_{i+2}$ plus random samples S_r are encoded with the transformer masked language model. The encoded representation of the anchor is used to predict each target at its target distance. The S_i objects are raw text sentences, the *encoder* is the transformer model, and c_i and t_i are vectors.

(Vaswani et al., 2017). Cross-attention is the mechanism in neural networks that allows for attention to be shared between multiple inputs, in our case, two separate spans of text.

Alternatively, we can encode the anchor and target together and then dot product the latent vector with a learned vector representation for each distance k. We call this approach a *uni-encoder*. With this encoder, equation 2 will be $f_k(s_{i+k}, s_i) = \exp(t_{i+k}^T w_k)$. The class matrix W_k in equation 2 is replaced by a class vector w_k, which has fewer parameters. This is similar to the ordering objectives in BERT and ALBERT where the pooled representation is used for a binary classification task and the learned vector representation for each distance k is just the softmax weights. The potential drawback to this method is that each pair of sentences is represented by a single vector. This encoder may learn a representation that is similar for all examples that have the same label but does not explicitly model the content of the input.

CONPONO implements the intersection of these two approaches. The targets are concatenated to the anchor when encoded, to make use of the cross-attention of the transformer encoder. The anchor, is encoded independently, though with the same weights. This objective allows for more freedom in the values of c_i and t_{i+k}, unlike the *uni-encoder*. Furthermore, since the encoder, g_θ, can encode either one span (s_i) or two spans (s_i, s_{i+k}), it can be used for downstream tasks that have either single (eg. SSP) or double (eg. BSO) span inputs.

2.2 Comparing Inter-Sentence Modeling Objectives

There are different *tasks* that can be used for learning inter-sentence representations. BERT (Devlin et al., 2019) included a next sentence prediction (NSP) task. For NSP, two spans are fed into the model with the second span either being the next contiguous span of text from the source or 50% of the time it is replaced with a random span from the corpus. The task is a binary classification of whether the two spans are from the same source. ALBERT (Lan et al., 2019) compares the NSP approach to using no inter-sentence objective and to sentence order prediction, which for clarity we refer to as binary sentence ordering (BSO). For BSO, the input is two spans that are always contiguous and from the same source but 50% of the time are in reverse order. With CONPONO we capture the benefits of both learning ordering between coherent sentences and contrasting against random negatives. We make the objective even more challenging by also predicting order on spans that are multiple sentences apart, and using other sentences from the same paragraph as harder negatives.

2.3 Technical details

In practice, we use a 512 token input which is much larger than most two sentence pairs. To train on longer sequence lengths, we use 4 sentences as the anchor and 3 sentences as the target segment. We truncate longer sentences and pad tokens up to the sequence length as done for typical BERT input. There is no overlap between the two segments and the k distance refers to the number of sentences omitted between the two segments. For example, for a paragraph we may choose $s_7..s_{10}$ as the anchor and $s_1..s_3$ as the target for $k = -4$ because s_3 is 4 positions behind s_7. Since most paragraphs are not long enough to have many sentences in both directions of a 4 sentence anchor, we randomly select 4 of the 8 possible k targets for a given paragraph. Because of the random sampling, we oversample shorter distances because they occur more consistently in the data.

We train with 32 input sentences, where 1 is the correct target, 3 are hard negatives from the same document and 28 are random sentences from other documents. For fair comparison, we train on the same data as BERT, using only Wikipedia and BooksCorpus (Zhu et al., 2015). We initialize our model with BERT-Base weights and train until the model has seen one-fourth as many segment pairs as the original BERT model (32M total), so the total compute and iterations of training are not significantly greater than BERT-Base. We also use a masked language model objective similar to BERT but dynamically mask during training for different masks each epoch. When jointly encoding two inputs, we concatenate the input tokens and separate the two spans with a "[SEP]" token to mimic the BERT format.

3 Evaluation

We evaluate our model on the DiscoEval benchmark (Chen et al., 2019) and on the RTE (Giampiccolo et al., 2007; Bentivogli et al., 2009), COPA (Roemmele et al., 2011) and ReCoRD (Zhang et al., 2018) datasets. We chose the DiscoEval benchmark because it is intended to evaluate a model's ability to represent the "role of a sentence in its discourse context". We also report results on RTE, COPA and ReCoRD because these tasks have a discourse or sentence ordering aspect to them but are not exclusively designed for discourse evaluation.

3.1 Discourse Evaluation

Tasks: DiscoEval (Chen et al., 2019) is a suite of tasks "designed to evaluate discourse-related knowledge in pretrained sentence representations". The benchmark is composed of seven tasks; four based on sentence ordering or coherence (Sentence position (SP), Binary sentence ordering (BSO), Discource coherence (DC) and Sentence section prediction (SSP)) and three that are based on classifying the type of relationship between a pair of text sequences (Penn Discourse Tree Bank Explicit and Implicit (PDTB-E/I) and Rhetorical structure theory (RST)). PDTB (Prasad et al., 2008) and RST (Carlson et al., 2001) are human annotated datasets. Both are multi-class classification tasks where PDTB is classifying a pair of sentences whereas RST is predicting the class of a node in a document-level discourse tree. Both classes of tasks are critical aspects of understanding discourse.

Baselines: The previously best overall performing model from DiscoEval (Chen et al., 2019) was BERT-Large (Devlin et al., 2019). We also include the results for BERT-Base because our model is most comparable to BERT-Base in terms of parameter size, training data and training compute. We also evaluate RoBERTa-Base (Liu et al., 2019) because it was trained on more data, reported improvements over BERT-Base on other tasks but dropped the next sentence prediction objective entirely. We also compare against a BERT-Base model which we trained with binary sentence ordering (BERT-Base BSO) because this objective has been shown to be more useful than next sentence prediction (Lan et al., 2019). This BERT-Base BSO model was initialized with BERT weights and trained on the same data but only on contiguous spans of text where 50% of the time we switch the order. This model and CONPONO are initialized from the same weights and trained on the same number of segment pairs so that the two models can be compared fairly.

In Section 2.1 we describe different encoding approaches for generating the sentence-level representations. We report results from versions of CONPONO using each of these encoding approaches, labeled *isolated* to represent separate encoding and *uni-encoder* to represent joint encoding of the anchor and target without a separate anchor encoding. The final line in Table 1 is the combined approach that we describe in Section 2.

Modeling DiscoEval We reuse the code from DiscoEval and generally maintain the same process for collecting our results on the benchmark, such as freezing all weights and only training a logistic regression or one layer perceptron on top of the sentence encodings. Note that since we are only interested in the vector representations of the input, we drop the weight matrix W_k and only use the output of the encoder. We omit the details for

Model	SP	BSO	DC	SSP	PDTB-E	PDTB-I	RST-DT	avg.
BERT-Base	53.1	68.5	58.9	80.3	41.9	42.4	58.8	57.7
BERT-Large	53.8	69.3	59.6	**80.4**	**44.3**	43.6	59.1	58.6
RoBERTa-Base	38.7	58.7	58.4	79.7	39.4	40.6	44.1	51.4
BERT-Base BSO	53.7	72.0	71.9	80.0	42.7	40.5	**63.8**	60.6
CONPONO *isolated*	50.2	57.9	63.2	79.9	35.8	39.6	48.7	53.6
CONPONO *uni-encoder*	59.9	74.6	72.0	79.6	40.0	43.9	61.9	61.7
CONPONO (k=2)	**60.7**	**76.8**	**72.9**	**80.4**	42.9	**44.9**	63.1	**63.0**
CONPONO std.	±.3	±.1	±.3	±.1	±.7	±.6	±.2	-

Table 1: CONPONO improves the previous state-of-the-art on four DiscoEval tasks. The average accuracy across all tasks is also a new state-of-the-art, despite a small drop in accuracy for PDTB-E. BERT-Base and BERT-Large numbers are reported from Chen et al. (2019), while the rest were collected for this paper. We report standard deviations by running the evaluations 10 times with different seeds for the same CONPONO model weights.

the encoding logic for each task since that is explained in detail in Chen et al. (2019). Here we only mention our deviations from the Chen et al. (2019) methodology. The most salient difference is that we only use the *pooled* representation from our model rather than the average from multiple layers of the model for the SP, BSO and DC tasks.

For encoding individual tasks we prefer to encode pairs of sentences together. For SP we encode the first sentence concatenated with every other sentence instead of taking the point-wise difference and concatenate the 5 vectors. For BSO we also encode the two sentences together instead of separately. For DC we split the paragraph into pairs of sentences and encode those together. We concatenate the 3 output vectors. For RST instead of embedding each sentence and doing a mean of all the sentences in a subtree, we simply concatenate those sentences and encode them all together as a single text span. Any text segments longer than 512 tokens are truncated from the end.

Results: Table 1 shows that our model outperforms the previous state-of-the-art accuracy on DiscoEval overall. Our model excels in particular on the sentence ordering and coherence tasks (SP, BSO, and DC). Note that our model parameter count is the same as BERT-Base but it outperforms BERT-Large, which has significantly more parameters and has used much more compute for pretraining. From the discussion in Section 2.2, BERT represents using the NSP objective and we train BERT-Base BSO to compare NSP, BSO and CONPONO directly. BERT-Base BSO scores tend to fall between those of BERT-Base and our model, implying that the sentence ordering objective is improving the models for this benchmark, but that binary sentence ordering is not sufficient to capture the added benefits of including more fine-grained ordering and negative examples.

We observe that CONPONO outperforms both the *isolated encoding* and *uni-encoding* approaches. CONPONO *isolated* preforms significantly worse than both other approaches, suggesting that cross-attention between the anchor and the target is critical to learning stronger discourse representations. CONPONO *uni-encoder* results are closer to our combined encoding approach but still fall short on every task. This empirical result suggests that the separate encoding of the anchor during pretraining is important despite the fact that theoretically CONPONO could trivially reduce to the *uni-coder* representation by ignoring c_i.

3.2 RTE, COPA and ReCoRD

Tasks: DiscoEval was specifically designed to evaluate model performance on discourse tasks but there are many other benchmarks that could also benefit from pretraining for improved discourse coherence. We evaluate our model on three such tasks, Recognizing Textual Entailment (RTE) (Giampiccolo et al., 2007; Bentivogli et al., 2009), Corpus of Plausible Alternatives (COPA) (Roemmele et al., 2011) and Reading Comprehension with Commonsense Reasoning Dataset (ReCoRD) (Zhang et al., 2018). We report accuracy on the validation set provided by each dataset.

Each example in RTE is a pair of sentences. The model must classify whether or not the second sentence entails the first. Examples in COPA are composed of a single context sentence followed by two candidate sentences that are either a cause or effect of the context sentence. The model must select the

Context	Completions
ReCoRD	
... Despite its buzz, the odds are stacked against *Google*'s *Chrome OS* becoming a serious rival to *Windows*... *Chrome OS* must face the same challenges as *Linux*: compatibility and unfamiliarity. A big stumbling block for *Google* will be whether its system supports *iTunes*.	Google will also be under pressure to ensure [**Chrome OS** / iTunes / Linux] works flawlessly with gadgets such as cameras, printers, smartphones and e-book readers.
RTE	
Rabies virus infects the central nervous system, causing encephalopathy and ultimately death. Early symptoms of rabies in humans are nonspecific, consisting of fever, headache, and general malaise.	**Rabies is fatal in humans.**
COPA	
The women met for coffee.	**They wanted to catch up with each other.** The cafe reopened in a new location.

Table 2: These are examples from ReCoRD, RTE, and COPA that exhibit aspects of discourse coherence. For ReCoRD, candidate entities are in italics and replaced terms in the completion are underlined. True completions are bold.

most "plausible" sentence of the two. Lastly, an example in ReCoRD is a paragraph from a news article, followed by several bullet points and with all the entities marked. The model is given a single sentence from later in the document with a single entity masked out and must select the entity from the context that fills the blank. Table 2 shows examples of each with correct choices in bold.

Baselines: We compare our model against BERT-Base because this is the closest model in terms of parameter size and training data. However, since our model is initialized with BERT-Base weights, we also report results from BERT-Base BSO because it was trained on the same number of text examples as Conpono. We also compare against BERT-Large to contrast to a much larger language model. We provide results from Albert (Lan et al., 2019) when available to provide a state-of-the-art baseline that may have used more data, compute and parameters. The purpose of these results is not to compare against the current state-of-the-art but rather to better understand the improvements that can be found from adding a discourse coherence objective to BERT-Base without significantly increasing the model size or training data.

Results: We believe that the coherence and ordering aspects of these evaluation tasks are well fit to demonstrate the how our model can improve on strong baselines such as BERT-Base. Table 3 shows that our model achieves accuracies on RTE and COPA comparable to BERT-Large while having the same number of parameters as BERT-Base. Interestingly, we observe improvements over the baseline with BERT-Base BSO, showing that even simple discourse-level objectives could lead to noticeable downstream effects. Though these improvements are modest compared to BERT-Large, they are meant to highlight that our model does not only improve on results for artificial sentence ordering tasks, but also on aspects of benchmarks used to generally evaluate pretrained language models and language understanding.

Model	RTE	COPA
BERT-Base	66.4	62.0
BERT-Base BSO	71.1	67.0
Conpono	70.0	69.0
BERT-Large	70.4	69.0
ALBERT	86.6	-

Table 3: Our model improves accuracy over BERT-Base for RTE and COPA benchmarks. Improvements are comparable to BERT-Large but still lag behind much larger models trained on more data, such as ALBERT. All scores are on the validation set.

3.2.1 ReCoRD results and models

Model	Accuracy
BERT-Base	61.2
Conpono	63.2
BERT-Large	69.8 [EM]

Table 4: Conpono is more effective at classifying the most plausible sentence from the extended context than BERT-Base. We report the BERT-Large exact match score, where the model selects only the target entity from the context, for reference. All scores are on the validation set.

The task for the ReCoRD dataset is to select the correct entity from those that appear in the context to fill in the blank in the target. Previous models for ReCoRD have used a similar structure to SQuAD (Rajpurkar et al., 2016) where the model outputs a vector for each token and the model learns the best start and end position of the answer span based on the softmax over all the tokens. We, instead, generate all possible target sentences by filling the blank with each marked entity and discriminatively choose the sentence most likely to be the true "plausible" sentence from the context. This modified task evaluates how our model compares to BERT-Base choosing the most coherent sentence from a set of nearly identical sentences. In Table 4 we show that CONPONO does achieve a boost over BERT-Base but is still well below BERT-Large exact match score on the harder task of selecting the entities in context. The strong results from BERT-Large imply that having a better representation of the text with a large model is able to subsume any improvement from learning plausible contexts for this task.

3.3 Ablations

There are three aspects of our modeling choices that warrant a deeper understanding of their importance to the model:

- *Window size:* We ablate the 4 window sizes (ie. choices of k). k = 1 is effectively binary sentence ordering with negative samples.
- *Masked Language Model Objective:* We remove the MLM objective allowing the model to optimize only the CONPONO objective without maintaining a good token level representation.
- *Model size:* We train a smaller model that is also initialized with pretrained weights.

To measure the effects of each of these design decisions, we report DiscoEval scores for each model as well as accuracy on the CONPONO classification task on a held-out set of examples. This is to show how well the model is optimized as well as how well it performs on downstream tasks.

Table 5 shows the results on DiscoEval with our model and several key ablations. We observe that using a window size for our objective that is larger than 1 is key to seeing downstream improvements. We believe that this is due to the objective being harder for the model because there is more variation farther from the anchor. At the same time, increasing the window size beyond 2 seems to result in similar performance. This may be because larger distances from the anchor also lead to more ambiguity. We see this reflected in the held-out classification accuracy being lower for examples with larger distance labels in Figure 2.

We also note that keeping the masked language model objective during pretraining also improves downstream performance. In Figure 2 we see that classification accuracy is consistently lower with the MLM objective compared to without. This is expected because during inference, many key terms may be masked out, making the task harder. However, keeping this objective during pretraining maintains a good token-level representation that is necessary for downstream tasks.

Lastly, we try training a smaller version of our model, with only 2 hidden layers, and a 512 intermediate size. The smaller model is able to train much faster, allowing us to train on many more examples and new data. However, we are unable to achieve similar results despite training on 24 times more examples, and including CCNews (Liu et al., 2019), a larger and higher quality data source.

3.4 Qualitative Analysis

To glean some insight into how CONPONO representations may differ from BERT-Base representations, we look at the occurrence of discourse markers in the BSO-Wikipedia task of DiscoEval. We choose this task because it is a simple binary classification task that has only 2 sentences as input and the domain is similar to the pre-training data. We look at the usage of discourse markers identified by Nie et al. (2017); *but, when, if, before, because, while, though, after, so, although, then, also, still.* [3]

We extract examples from the test set on which CONPONO output the correct label and BERT-Base output the incorrect label and visa versa. For each set of examples, we measure the change in the occurrence of discourse markers relative to the training data counts. Since some markers are much more common than others, we take the weighted average of the change in appearance rate, where the weights are the training data counts of each marker.

[3] We omit *and* and *as* because they are very common in this corpus but often are not used as connectives between the two candidate sentences for the BSO task.

Model	SP	BSO	DC	SSP	PDTB-E	PDTB-I	RST-DT	avg.
k=4	59.84	76.05	**73.62**	**80.65**	42.28	44.25	63.00	62.81
k=3	60.47	76.68	72.74	80.30	**43.40**	44.28	62.56	62.92
k=2	**60.67**	**76.75**	72.85	80.38	42.87	**44.87**	**63.13**	**63.07**
k=1	47.56	66.03	72.62	80.15	42.79	43.55	62.31	59.29
- MLM	54.92	75.37	68.35	80.2	41.67	43.88	61.27	60.81
Small	45.41	61.70	67.71	75.58	35.26	36.18	46.58	52.63

Table 5: The ablation analysis shows the effects of different k values (ie. window sizes) in our objective, removing the MLM objective during pretraining and training with a small transformer encoder.

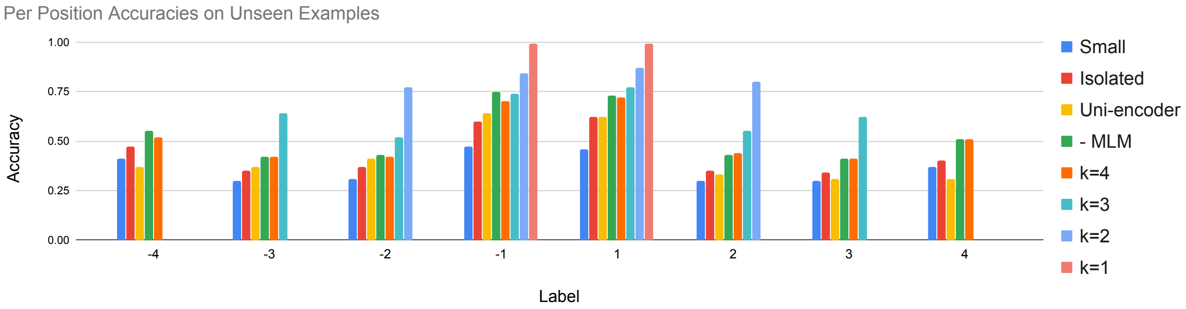

Figure 2: We can evaluate the accuracy on the CONPONO objective for each label (ie. distance between anchor and target sentence) on a set of 5,000 examples held-out from training. We observe that higher accuracy does not necessarily correlate with better downstream performance on DiscoEval.

We find that in the set of examples that CONPONO classified correctly, the rate of discourse makers was 15% higher than in the training corpus. This is in contrast to 11% higher among the examples that BERT classified correctly. The standard deviation for random samples of the same size was about 1%. This suggests that both BERT and CONPONO are relying heavily on discourse markers to solve the BSO-Wikipedia task.

While it is expected for shallow discourse markers to be strong features for sentence ordering, we expect CONPONO to also incorporate deeper features, such as anaphora, due to its pretraining objective. One indication of CONPONO relying on alternative features than BERT-Base is that there was a 12% relative increase in discourse markers in the CONPONO set when counting markers only in the first sentence whereas an 8% relative increase in the BERT set when counting markers only in the second sentences. The difference in the location of the discourse markers in the two sets of examples suggests that CONPONO and BERT utilize those features differently and that CONPONO may be less likely to incorrectly classify examples that use discourse markers in the first sentence of a BSO example. Manually inspecting a sample of examples hints that there are often strong coreferences between the two input sentences that indicate the ordering.

Table 6 shows two examples from the CONPONO correct set which is drawn from the BSO-Wikipedia test data. In both examples, the discourse marker appears in the first sentence but the second sentence contains anaphora referring to an antecedent in the first sentence.

4 Related Work

Some of the largest improvements on benchmarks such as GLUE (Wang et al., 2018) have come from ELMO's large scale bi-directional language modeling (Peters et al., 2018), BERT's masked language models (Devlin et al., 2019), XLNET's generalized autoregressive pretraining (Yang et al., 2019), RoBERTa's robust training (Liu et al., 2019) and ALBERT's parameter reduction techniques (Lan et al., 2019). As discussed in Section 2.2, most language model were limited to NSP or BSO for inter-sentence representation learning. We showed that by comparing to BERT, which uses NSP and BERT-Base BSO which we train with the BSO objective that our objective is able to improve the discourse-level representations by training on more fine-grained sentence ordering, non-contiguous

In 1941 [1]Vaughn joined the United States National Guard for what had been planned as a one-year assignment , **but when** [2]World War II broke out , he was sent abroad until the war ended in 1945 .
[1]He decided to make music a career when he was discharged from the army at the end of [2]the war , and attended Western Kentucky State College , now known as Western Kentucky University , majoring in music composition .
Although it lasted only twenty-three years (1933–1956) and enrolled fewer than 1,200 students , Black Mountain College was one of the most fabled experimental institutions in art education and practice .
It launched a remarkable number of the artists who spearheaded the avant-garde in the America of the 1960s .

Table 6: Two examples from the DiscoEval BSO-Wikipedia test set on which CONPONO made the correct prediction but BERT-base did not. **Bold** terms are discourse markers, underlined terms are co-referents. In both examples, the discourse marker appears in the first sentence but the second sentence has anaphora referring to an antecedent in the first sentence.

neighboring sentences and contrasting against random negatives.

Early approaches to sentence representation, such as Skip-Thought Vectors (Kiros et al., 2015), mimicked word embedding methods in addition to left-to-right language modeling to use unlabeled data to learn sentence level representations. DisSent (Nie et al., 2019) focused more on collecting data that could be used to train a supervised classification model on pairs of sentences. These and other innovations in sentence representation lead to the creation of more evaluations for discourse and coherence representation (Chen et al., 2019; Xu et al., 2019).

Like other unsupervised representation learning models, CONPONO is trained to generate a latent variable that encodes inter-sentence relationship and discourse coherence. Our objective is inspired by the Contrastive Predictive Coding (CPC) objective (van den Oord et al., 2018). CPC was originally designed to be a "universal unsupervised learning approach to extract useful representations from high-dimensional data" and was previously implemented on the token-level for text models. We utilize the k-distance predictions of CPC because it naturally captures discourse and sentence ordering properties when applied on the sentence-level. Furthermore, by combining our objective with a transformer encoder, our model is able to benefit from cross-attention between the anchor and the target sentences, which we show outperforms encoding the anchor and target separately, as implemented in CPC. In Section 3.3 we show that the cross-attention is an important factor in learning a good representation for downstream tasks and effectively optimizing our inter-sentence objective.

5 Discussion

In this paper we present a novel approach to encoding discourse and fine-grained sentence ordering in text with an inter-sentence objective. We achieve a new state-of-the-art on the DiscoEval benchmark and outperform BERT-Large with a model that has the same number of parameters as BERT-Base. We also observe that, on DiscoEval, our model benefits the most on ordering tasks rather than discourse relation classification tasks. In future work, we hope to better understand how a discourse model can also learn fine-grained relationship types between sentences from unlabeled data. Our ablation analysis shows that the key architectural aspects of our model are cross attention, an auxiliary MLM objective and a window size that is two or greater. Future work should explore the extent to which our model could further benefit from initializing with stronger models and what computational challenges may arise.

Acknowledgments

We wish to thank the Stanford NLP group for their feedback. We gratefully acknowledge support of the DARPA Communicating with Computers (CwC) program under ARO prime contract no. W911NF15-1-0462

References

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth PASCAL recognizing textual entailment challenge.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurovsky. 2001. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*.

Mingda Chen, Zewei Chu, and Kevin Gimpel. 2019. Evaluation benchmarks and learning criteria for discourse-aware sentence representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 649–662, Hong Kong, China. Association for Computational Linguistics.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Allen Nie, Erin Bennett, and Noah Goodman. 2019. DisSent: Learning sentence representations from explicit discourse relations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4497–4510, Florence, Italy. Association for Computational Linguistics.

Allen Nie, Erin D Bennett, and Noah D Goodman. 2017. Dissent: Sentence representation learning from explicit discourse relations. *arXiv preprint arXiv:1710.04334*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

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, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn discourse TreeBank 2.0. In *LREC 2008*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Peng Xu, Hamidreza Saghir, Jin Sung Kang, Teng Long, Avishek Joey Bose, Yanshuai Cao, and Jackie Chi Kit Cheung. 2019. A cross-domain transferable neural coherence model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 678–687, Florence, Italy. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018. ReCoRD: Bridging the gap between human and machine commonsense reading comprehension. *arXiv preprint 1810.12885*.

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 *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

A Appendix

We include some fine-grained DiscoEval results that were reported as averages, as well as implementation and reproduction details for our experiments.

A.1 SP, BSO and DC breakdown

Table 7 shows the scores for each model per each dataset domain for the SP, BSO and DC tasks in DiscoEval.

A.2 CONPONO pretraining details

CONPONO is pretrained on 1.6 million examples randomly sampled from Wikipedia and BooksCorpus. We use the same number of training examples for all the ablations and training BERT-Base BSO. On example consists of a single anchor and 32 candidate targets, 4 losses (1 for each of the 4 randomly chosen true targets (ie. k)). We use a 25% warm up rate and a learning rate of 5e-5. The model is initialized with BERT-Base weights. We add a square interaction weight matrix that is the same size as model output dimensions (ie. 756) that is referred to as W_k in Section 2. There is one such matrix for each k. The maximum sequence length of the input is 512, though do to some preprocessing constraints, the maximum input seen by the model is 493.

Our CONPONO small model has a hidden size of 128, an intermediate size 512, and has 2 hidden layers. We train it on 38.4 million examples, including examples from CCNews. Samples are drawn from each source proportional to the size of the source, meaning that about 70% of training examples come from CCNews. Otherwise, we use all the same parameters as CONPONO.

A.3 Parameter counts

Table 8 shows the number of parameters in each model used.

A.4 RTE, COPA and ReCoRD details

RTE is trained for 3240 steps, with checkpoints every 750 steps and a learning rate of 8e-6. The warm-up proportion is 10% and the a maximum sequence length of 512

COPA is trained for 300 steps, with checkpoints every 50 steps and a learning rate of 1e-5. The warm-up proportion is 10% and the maximum sequence length of 512.

ReCoRD is trained for 8 epochs over the training data with a learning rate of 2e-5, warm-up proportion of 10% and a maximum sequence length of 512.

Model	Parameters
BERT-Base	110M
RoBERTa-Base	110M
Conpono [All Variants]	110M
BERT-Large	335M

Table 8

	SP			BSO			DC	
Model	Wiki	arxiv	ROC	Wiki	arxiv	ROC	Wiki	Ubuntu
BERT-Large	50.7	47.3	63.4	70.4	66.8	70.8	65.1	54.2
RoBERTa-Base	38.35	33.73	44.00	60.19	55.16	60.66	62.80	53.89
BERT-Base BSO	49.23	50.92	60.80	74.67	68.56	72.22	88.80	56.41
Conpono - MLM	50.95	51.90	61.92	77.98	71.45	76.68	86.70	50.00
Conpono Small	44.90	41.23	50.10	65.03	58.89	61.19	78.10	57.32
Conpono *isolated*	49.33	44.60	56.53	59.16	57.48	56.94	71.60	54.71
Conpono *uni-encoder*	54.30	58.58	66.75	78.25	71.65	73.99	86.00	57.90
k=4	54.07	58.30	67.15	79.04	72.21	76.89	88.38	**58.85**
k=3	54.65	**59.55**	67.22	**79.34**	**73.61**	**77.08**	**89.48**	56.00
k=2	**54.83**	58.77	**68.40**	79.24	74.16	76.84	89.22	56.41
k=1	44.05	40.98	57.65	68.47	62.40	67.24	89.03	56.20

Table 7: SP, BSO and DC are composed of separate datasets. We report the average in the main paper but show the breakdown here.