Fortunately, Discourse Markers Can Enhance Language Models for Sentiment Analysis

Liat Ein-Dor[*1], **Ilya Shnayderman**[*1], **Artem Spector**[*1], **Lena Dankin**[1],
Ranit Aharonov[†1] **and Noam Slonim**[1]

[1]IBM Research
{liate,ilyashn,artems,lenad,ranit.aharonov2,noams}@il.ibm.com

Abstract

In recent years, pretrained language models have revolutionized the NLP world, while achieving state of the art performance in various downstream tasks. However, in many cases, these models do not perform well when labeled data is scarce and the model is expected to perform in the zero or few shot setting. Recently, several works have shown that continual pretraining or performing a second phase of pretraining (inter-training) which is better aligned with the downstream task, can lead to improved results, especially in the scarce data setting. Here, we propose to leverage sentiment-carrying discourse markers to generate large-scale weakly-labeled data, which in turn can be used to adapt language models for sentiment analysis. Extensive experimental results show the value of our approach on various benchmark datasets, including the finance domain. Code, models and data are available at https://github.com/ibm/tslm-discourse-markers.

1 Introduction

Large pretrained language models are reshaping the landscape of NLP. These models, recently referred to as *foundation models* (Bommasani et al. 2021), were originally proposed with a two-step paradigm in mind. The model is first pretrained at scale on broad data with a surrogate self-supervised task; the knowledge gained by this pretraining is then transferred and adapted via fine-tuning on – typically small – labeled data, to a specific downstream task. Prominent examples include BERT (Devlin et al. 2019) and GPT-3 (Brown et al. 2020). The practical value of this approach is immense. The self-supervised pretraining requires no labeled data. The resulting model represents a single powerful starting point that can be swiftly adapted to address a wide range of target tasks with relatively little annotation effort, via few-shot or even zero-shot learning (Brown et al. 2020).

Subsequent studies have shown that the original two-step paradigm can be further refined to yield an even better starting point model for particular tasks of interest. For example, continuing the pretraining on domain-specific data such as

finance or legal documents have proven beneficial to tasks in these domains (Araci 2019; Chalkidis et al. 2020; Gururangan et al. 2020). Similarly, additional pretraining of BERT on dialog data yields better results in target tasks related to dialogue application (Wu et al. 2020), and continual pretraining of BERT on product reviews with sentiment-aware pretraining tasks led to improved performance in sentiment analysis in this domain (Zhou et al. 2020). Another, more computationally demanding option is to pretrain the model from scratch on self-supervised task(s) that aim to better reflect the nature of the target tasks. For example, the pretraining tasks of SpanBERT (Joshi et al. 2019) and PEGASUS (Zhang et al. 2020a) are designed to be closer in spirit to span-extraction tasks as in question answering and to summarization tasks, respectively, resulting in better performance in these target tasks.

A related path, which is further explored in this work, is to add an intermediate training step, referred to as *inter-training*, which is somewhat aligned with a specific target task of interest. There are several aspects by which these inter-training approaches differ. One main aspect is the similarity between the intermediate task and the target task which ranges from full alignment using weakly or readily available labeled data (Meng et al. 2020; Zhou et al. 2020; Huber et al. 2021) to transfer learning using labeled data on a similar yet different task (Pruksachatkun et al. 2020), and further including works which perform transfer learning with no labeled data, e.g., (Shnarch et al. 2021) apply unsupervised text clustering and then inter-train a model to predict the cluster label. Among the approaches that rely on fully aligned intermediate tasks, some works leverage weak labels that are inherent to the original text, like the explicit mention of the class name (Meng et al. 2020) or the presence of the token 'that' in a sentence (Levy et al. 2018); while others rely on non-textual signals like human-added numeric review ratings (Zhou et al. 2020) or sentiment-bearing emojis (LeCompte and Chen 2017). Weak labels that are inherent to the text usually have limited coverage and involve bias towards specific keywords or patterns that define the weak signal. While the non-textual signals usually do not suffer from these issues, since they are external to the text, they are often specific to task and domain and therefore are less directly applicable to new tasks and/or in new domains.

The present work suggests a new type of weak labels which

---

are inherent to the original text but at the same time can be perceived as an external label that can be removed from the original text while keeping the remaining text meaningful and grammatical (Moder and Martinovic-Zic 2004).

Specifically, we propose to leverage the signal carried by particular discourse markers (DMs) to generate large amounts of weakly labeled data for the important task of sentiment analysis (SA). For example, we assume that sentences following the prefixes "Happily," and "Sadly," convey a positive sentiment and a negative sentiment, respectively. Exploiting this simple assumption with a small seed of 11 discourse markers, we generate large amounts of weakly labeled data out of a large and general English corpus. Inter-training BERT-base and BERT-tiny on this data yields significant performance improvements, especially when labeled data is scarce and in a zero-shot scenario. Moreover, we show how to use the obtained classifier to automatically reveal sentiment-carrying discourse markers in particular domains. Relying on these domain-specific sentiment-carrying discourse-markers yields an additional performance gain in zero-shot learning, and may further open the door for additional future applications. In summary, our main contributions are:

1. A novel approach that leverages sentiment signals of discourse markers for creating sentiment-aware language models that significantly outperform prior models.

2. A new method for enhancing domain-specific sentiment classification, based on statistical analysis of discourse markers in a domain-specific corpus.

3. A large dataset of weakly labeled sentences from Wikipedia, and a code for generating weakly labeled data from a given text corpus.

2 Related Work

Learning with Discourse Markers Discourse markers (DMs) are words or phrases that play a role in managing the flow and structure of discourse. DMs have been used as a learning signal for the prediction of implicit discourse relations (Liu and Li 2016; Braud and Denis 2016) and inference relations (Pan et al. 2018). The task of DM prediction has been leveraged in several works such as (Jernite, Bowman, and Sontag 2017; Nie, Bennett, and Goodman 2019; Sileo et al. 2019), to learn general representations of sentences, which can be transferred to various NLP classification tasks. Sileo et al. were the first to systematically study the association between *individual* DMs and *specific* downstream task classes. Using a model trained to predict discourse markers between sentence pairs, they predict plausible markers between sentence pairs with a known semantic relation (provided by existing classification datasets). Based on these predictions, they study the link between discourse markers and the semantic relations annotated in classification datasets.Here we show how such an association can be leveraged to enhance the performance of language models on a downstream task, and furthermore in a particular domain.

Task-aware Language models. A recent line of works has been focused on bridging the gap between the self-supervision task and the downstream tasks which is inherent to multi-purpose pretrained models (Sun et al. 2019; Tian

et al. 2020; Chang et al. 2020). In (Joshi et al. 2020), spans of texts are masked rather than single tokens, resulting in a language model oriented to span-selection tasks. Chang et al. suggested a language model targeted at document retrieval, and Zhang et al. pursued a similar goal for abstractive text summarization. For sentiment analysis, several works have incorporated sentiment knowledge into the pretraining task (Tian et al. 2020; Gu et al. 2020), while focusing mainly on word-level sentiment prediction tasks. Here, in order to achieve full alignment with the downstream task of sentence-level sentiment classification, we suggest a model that incorporates a *sentence-level* sentiment prediction objective. Similar objective was used in (Zhou et al. 2020), relying on ratings as sentiment signals, which are specific to the reviews domain. In contrast, our approach relies on sentiment signals that are carried by discourse markers, which are an inherent to language itself and are therefore available for a wide range of domains.

3 $SenDM$: A New Sentiment Language Model

3.1 Training DM-based Sentiment Models

We propose a general approach to develop DM-based sentiment models, and use it to develop $SenDM$, a new sentiment language model. The models rely on a large weakly labeled sentiment data set, which is automatically derived from a given corpus by leveraging strong associations between DMs (discourse markers) and sentiment classes, as depicted in figure 1. Given a corpus C and a list L of DMs that signal either a positive or a negative sentiment, each accompanied by its class label, we follow the heuristic introduced by (Rutherford and Xue 2015) and look for all sentences in C that start with $l \in L$ followed by a comma. We then remove l and the comma from the beginning of each sentence, and annotate all resultant sentences with the class label associated with l. This process results in a binary classification dataset for sentiment analysis, which is used to fine-tune a pre-trained language model, M (*inter-training*). In this work we use the above flow to generate $SenDM$, and also to build an additional domain-adapted model as will be discussed in section 4.

3.2 The $SenDM$ model

We introduce $SenDM$, a general sentiment model, that aims to improve the performance of sentiment classification across domains. $SenDM$ is obtained using the flow described above, where C is a general corpus of newspaper and journal articles, denoted C_g (see section 3.3), and L is a seed list of sentiment related DMs obtained manually using general knowledge of the English language. More specifically, we asked 3 annotators, to go over a list of 173 commonly used DMs described in (Sileo et al. 2019), and mark any DM as positive/negative if it is likely to open a sentence bearing a positive/negative sentiment, based on its common usage in the English language. The final list, L_g, consists of 11 DMs, selected by all 3 annotators. The DMs identified as associated with a positive sentiment are: *'luckily', 'hopefully', 'fortunately', 'ideally', 'happily', and 'thankfully'*. Those associated with a negative sentiment are: *'sadly', 'inevitably',*
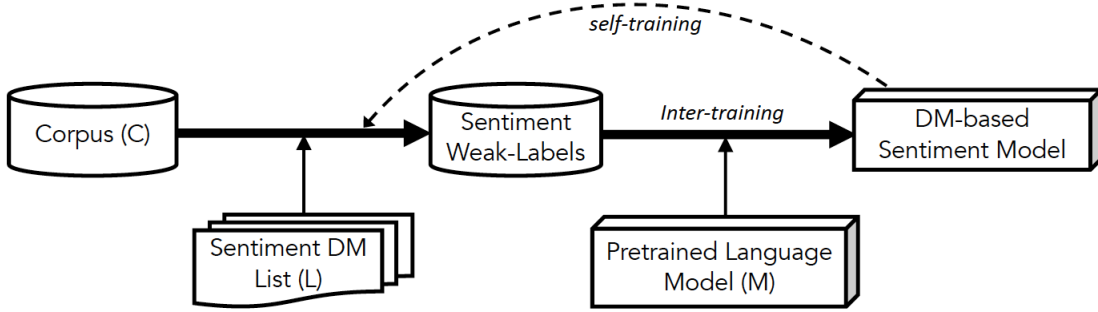
Figure 1: Overview of how DM-based sentiment models are trained.

'unfortunately', 'admittedly', and 'curiously'. The resulting weakly labeled data is used to fine tune both the uncased base and tiny architectures of BERT (Devlin et al. 2019; Jiao et al. 2020); We denote the resulting models by SenDM-base and SenDM-tiny, respectively, and release both of these models as part of this work.

3.3 Experimental Setup

The General Corpus (C_g) Our proposed solution relies on the availability of a corpus of unlabeled text. We use a corpus of some 400 million newspaper and journal articles[1], breaking the articles into sentences, and indexing these sentences. We focus on English sentences [2] and following (Sileo et al. 2019) we use only sentences which are $3 - 32$ tokens in length and have balanced parentheses.

Inter-training Details. The inter-training step (Figure 1) consists of fine-tuning BERT using weakly labeled data. For inter-training $SenDM$, we obtain a total of $1,876,614$ weakly labeled sentences, by using the list of sentiment-related DMs, L_g, over sentences in C_g, as described in section 3.1. We divide the samples into training (80%), development (10%), and test (10%) sets. We set the learning rate to $5e-5$, and the batch size to 32. We use the early stopping strategy, setting the max number of epochs to 4 and selecting the model with the best accuracy on the development set. The dropout probability is always kept at 0.1. We employ an Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e-06$. Training is performed on two V100 GPUs.

3.4 Evaluation Details

Evaluation is performed on the datasets appearing below, in three scenarios: zero-shot, few-shot and full-data. For zero shot we simply use the classification layer obtained from inter-training. For few-shot, we further fine-tune the inter-trained model with a small sample of n examples from the training set, with n ranging from 16 to 1024. We repeat each experiment five times with different random seeds, each time selecting different examples for fine-tuning. In the full-data scenario, all training examples are used for fine-tuning.

---

[1]From the LexisNexis 2011-2018 corpus,https://www.lexisnexis.com/en-us/home.page

[2]Specifically, sentences with probability $> 75\%$ of being English, based on Fast-Text langid from (Grave et al. 2018).

To support training on small samples, the batch size is set to 16. The other hyper-parameters are the same as in the inter-training phase described above, with one exception. For the few-shot scenario, which represents a low resource setting, we assume that no development set is available for employing the early stopping strategy. Instead, we follow the observation in (Zhang et al. 2020c) that for small training data, more iterations help stabilize BERT results, and set the number of epochs to 10.

Datasets The datasets used for evaluation are presented in Table 1. All datasets contain sentences that are labeled for sentiment. amazon, sst2, and yelp consist of review sentences. fpb75 is comprised of sentences from financial news.Most of these datasets provide more than two possible labels, so we adjust the datasets for the binary sentiment classification task. Specifically, fpb75 contains sentences that are labeled as neutral, which we remove from the training and test sets. amazon and yelp contain five different labels that reflect the sentiment ratings of each sentence ("stars"). We leave only sentences with the lowest and highest scores, considering those as negative and positive labels, respectively. For fine tuning we use up to 1024 examples from the training set. For testing we use the entire test set.

Dataset	Domain	Test set size
amazon	Product reviews	2K
yelp	Business reviews	20K
sst2	Movie reviews	1821
fpb75	Financial news	691

Table 1: Datasets used for evaluation. References for the datasets are as follows, by the order appearing in the table: (Keung et al. 2020), (Zhang, Zhao, and LeCun 2015), (Wang et al. 2018), (Malo et al. 2014).

3.5 Results

We now evaluate the performance of both the base and tiny versions of the general sentiment model (SenDM-base and SenDM-tiny) on datasets from different domains. Since our main focus is on the zero and few-shot setting, we report the results after fine tuning over 0 up to 1024 training examples.
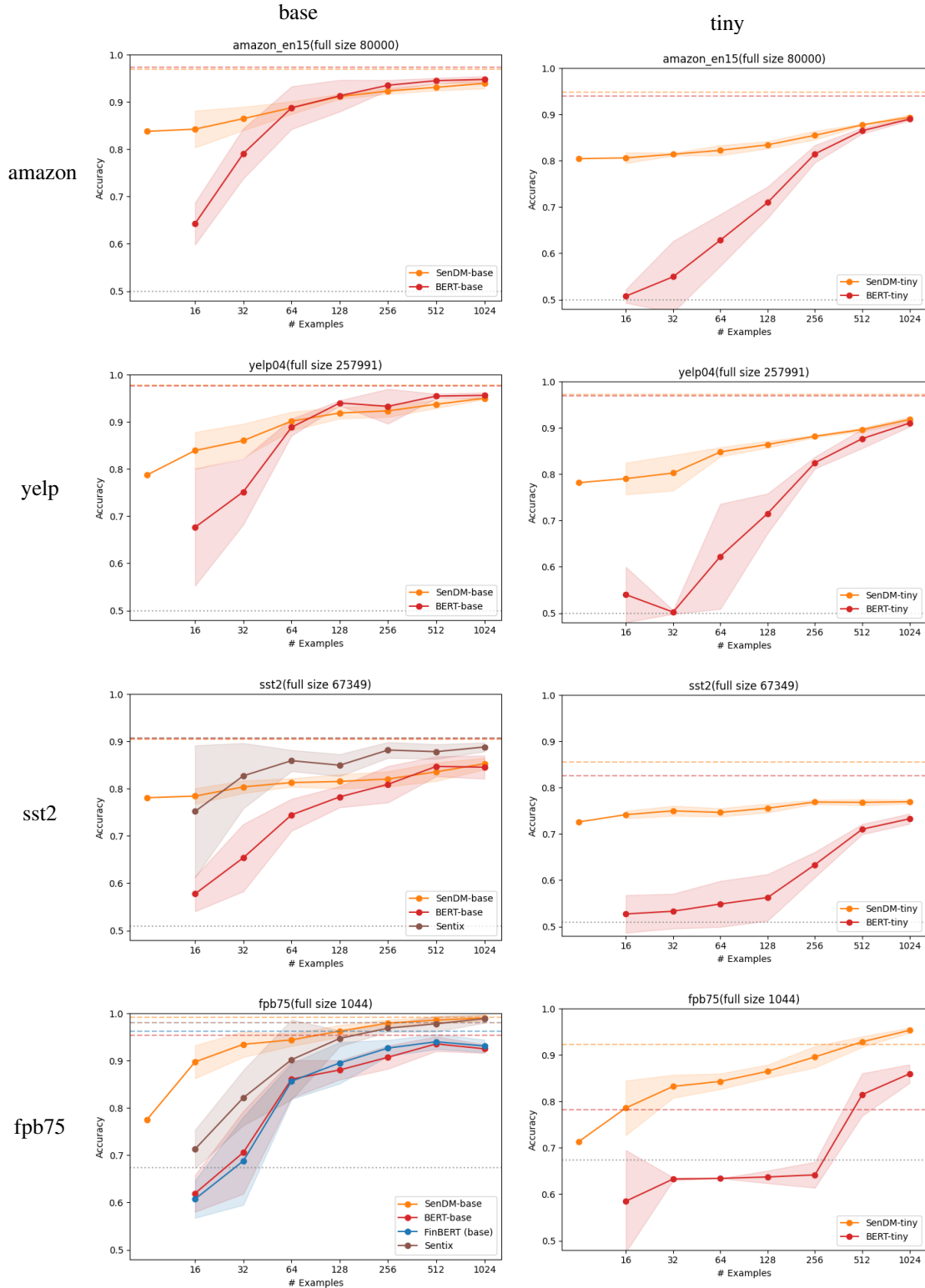
Figure 2: Performance of $SenDM$ and baselines on four datasets given different amounts of training examples. Left column: base-size models. Right column: tiny-size models. Lines indicate the mean and shaded areas indicate the standard deviation over the five seeds (see section 3.4 for details). Dashed horizontal lines indicate the fine-tuning results for the full training data (the full-data setting). Dotted horizontal lines indicate the prior of the common class in the dataset. FinBERT and SentiX are available only in the base size.

Figure 2 shows the accuracy of SenDM-base (left column) and SenDM-tiny (right column), for all datasets, vs. the number of examples used for fine tuning. The accuracy is compared to that of vanilla BERT – base and tiny respectively, and to SentiX (Zhou et al. 2020), that are fine tuned over the same labeled examples. For fpb75, we add the corresponding domain specific version of BERT-base – FinBERT (Araci 2019).

In all datasets, $SenDM$ significantly outperforms the BERT baselines, including the finance-specific FinBERT, especially when the number of examples used for fine tuning is relatively small. The gain in performance is even more significant when focusing on the tiny architecture. This is especially evident in the fpb75 dataset, where BERT-tiny completely fails to learn with up to 256 examples, whereas the SenDM-tiny is able to learn with as few as 16 examples. A similar trend can be seen for the sst2 dataset. As expected, the gap between $SenDM$ and its counterparts decreases with the increasing number of training examples, reflecting the decaying effect of the initial weights on the fine-tuned model. In most datasets, this gap completely vanishes in the full-data scenario, with the exception of fpb75, where the full training-data is of size 1044, only slightly larger than 1024 which is the upper limit of our few-shot regime.

From the stability perspective, $SenDM$ is more robust to changes in the initial seed compared to the other models, due to the lack of randomness in the initialization of its classification layer.

SentiX is a sentiment-aware pre-trained language model that was originally designed for cross-domain review sentiment analysis. Importantly, SentiX is trained on large amounts of Yelp and Amazon reviews, along with their associated star rating, the same star rating used to define the training set and the test set in our amazon and yelp datasets. Thus, one can not report zero/few-shot training results for this model in these two datasets, since the available model is already trained on large amounts of the respective train data. That said, it is intriguing to explore the performance of this model over our two other datasets. When considering the results in sst2, which is based on movie reviews, we see strong performance for SentiX. This is expected, since this kind of data, composed of starred reviews – albeit from a different domain – is precisely the forte of SentiX. Interestingly, though, its gap compared to our $SenDM$ is relatively small, and insignificant when fine tuning over 16 and 32 examples. Considering the results in a more distant domain, namely the fpb75 dataset, where starred reviews are irrelevant, we see the clear value of our approach, that consistently outperforms all other models, including SentiX, typically by a significant margin, especially when labeled data is scarce. These results support our hypothesis that pretraining based on sentiment-related DMs will result in a more robust model, that yields superior performance when tested on various domains.

A concern may arise, that the strong performance demonstrated by our approach on fpb75is related to the fact that the general corpus giving rise to the weak labels used for inter-training, also contains some financial documents, and that the results will be inferior for domains not covered in the corpus we start with. To address this concern, we generate a version of $SenDM$, in which financial documents are removed from the general corpus[3]. We find that there is no deterioration of results, supporting the notion that the observed improvement over alternative methods is not due to inter-training using financial documents – see Figure 1 in the Appendix.

Another concern we wanted to examine is related to the relevance of our approach for low resource languages, where a very large corpus like C_g is not available. To this end, we checked the sensitivity of the results to the size of the weakly-labeled data, by creating two versions of C_g, one based on inter-training using only 10%, and the other based on only 1%, of the weakly labeled data. Surprisingly, these two models resulted in no detrimental effect on the results. In addition, one may also leverage the large English weakly labeled data for inter-training the multilingual BERT model (M-BERT). We leave the examination of this approach for future research.

To summarize, overall, the proposed DM-based sentiment model, significantly improves sentiment classification performance for both small and large language models. Remarkably, even the tiny version of $SenDM$ outperforms the much larger BERT-base baseline.

4   Adapting $SenDM$ to a New Domain

In section 3.5 we saw that $SenDM$, which leverages a general list of sentiment-related DMs, improves results over baselines on all datasets, including the finance dataset, fpb75. Here we investigate whether adapting $SenDM$ to a new domain, can further improve its performance on that domain. We choose to study this on the financial domain since as stated in (Araci 2019), financial sentiment analysis is a challenging task due to the specialized language and lack of domain-specific labeled data. Moreover, it is an important task for many potential users, and finally, the adaptation impact can be tested given the availability of the fpb75 dataset.

4.1   The Training Approach

The robustness of $SenDM$ presumably emerges from the multi-domain corpus it relies on, and the general nature of the DM list, L_g, composed of discourse markers that are abundantly used and carry a general sentiment signal. However, due to potential domain-specific jargon and language style, given a domain specific text corpus C_d, it may be useful to build domain specific sentiment models.

We study five ways to build domain specific sentiment models based on the general flow described in Figure 1. All five resulting models, described in the bottom part of Table 2, are based on weakly labeled sentences from the domain-specific corpus, C_d. All five models rely on the availability of a general sentiment model, trained in a domain-independent manner, such as $SenDM$, which we release to the community. In the experiments we describe here, we use a variant of $SenDM$, denoted below $SenDM^*$, which is developed as $SenDM$, but after removing finance-related documents from the general corpus C_g, to better simulate the finance

---

[3]Based on topic tagging, see details in section 4.2.

domain as a new domain.[4] This $SenDM^*$ model is used as the straining point for inter-training all five models, and in some cases to define the inter-training weakly labeled data, as described next.

The first model, $SenDM_d^{L_g}$, uses the general DM list, L_g, for weak label extraction from text in the target domain. However, since sentiment-related DMs might be domain specific we develop a method to extract a domain-specific sentiment-related DM list, L_d. To that end, we note that there is no need to define a DM using the standard linguistic definition, and a functional definition can be used instead. Thus, we define as a sentiment-related DM , any n-gram ($n <= 3$) followed by a comma, for which the set of sentences it opens is enriched with highly confident positive/negative predictions, as determined by $SenDM^*$. The second model, $SemDM_d^{L_d}$, relies on a list composed of such DMs instead of L_g. As a third approach, we perform one step self-training, where the high confidence predictions of $SenDM^*$ over sentences from C_d are used for inter-training, ignoring the DMs.[5] This model is denoted by $SenDM_d^P$. Finally, aiming to reduce labeling noise, we explore a synergistic approach, i.e., taking as weakly labeled positive/negative sentences only those for which an opening DM conveys a sentiment signal, and that sentiment is consistent with the high confidence prediction of $SenDM^*$. We study this approach with our two DM lists, L_g and L_d, giving rise to two additional models, denoted $SenDM_d^{L_g+P}$ and $SemDM_d^{L_d+P}$, respectively.

Model name	C	L	M	With self-training
$SenDM$	C_g	L_g	BERT	NA
$SenDM_d^{L_g}$	C_d	L_g	$SenDM$	No
$SemDM_d^{L_d}$	C_d	L_d	$SenDM$	No
$SenDM_d^P$	C_d	NA	$SenDM$	Yes
$SenDM_d^{L_g+P}$	C_d	L_g	$SenDM$	Yes
$SemDM_d^{L_d+P}$	C_d	L_d	$SenDM$	Yes

Table 2: Sentiment language models and the corresponding assignment of C, L and M in the flow described in figure 1, as well as whether the predictions of $SenDM$ are used for assigning weak labels ("With self-training" – see main text for details) . $SenDM$ is a general (multi-domain) model. The other five are domain specific. C_g: a general, multi-domain, corpus; C_d: a corpus from domain d; L_g: a list of DMs associated with sentiment in the English language; L_d: such a list adapted to domain d.

Domain Specific Sentiment-Related DMs   We now turn to describe how we generate the domain specific sentiment-related DM list, L_d, given the domain specific corpus C_d and using $SenDM^*$. Note, we are not interested in identifying all domain specific sentiment related DMs. Rather, we are seeking a precision oriented list, for the purpose of obtaining a high-quality weakly labeled set of sentences for inter-training process. Thus, we perform strict automatic filtering in the process. The idea is to first identify all n-grams that may potentially serve as DMs, and then identify whether the set of sentences they open is enriched with positive/negative sentiment, based on the predictions of $SenDM^*$.

The first step consists of **identifying a list of candidate DMs**. To this end we first identify all unigrams, bigrams, and trigrams, that, followed by a comma, open sentences in C_d, and further group these using NER (e.g., instead of multiple bigrams of the type "on Sept 9th", "on 10/2/2020",... we generate one bigram "on DATE"). The candidate list consists of the 1000 most frequent DMs. Specific filters may be further applied depending on the domain corpora – see appendix for details.

The second step consists of **using SenDM* to select the domain-specific DMs out of the candidate list**. We analyse the sentences that start with the DMs in the above candidate list, to find those DMs whose associated sentences are significantly associated with a highly confident prediction of positive/negative sentiment. For each candidate DM, we sample 1000 sentences from the set of all sentences that start with the DM followed by a comma, and assign each of these sentences with a sentiment if it is scored with high confidence[6] by $SenDM^*$. For each candidate DM, we perform a statistical analysis of the sentiment of its associated sentences, on the sample of 1000 sentences, provided that they are not too repetitive based on token counts, and that the sentiment class with the higher count comprises at least 85% of the sentences assigned a sentiment. A DM is considered to be associated with a positive/negative sentiment if the p-value of the positive/negative class is smaller than 0.01 after Bonferroni correction for multiple tests, based on a Hypergeometric test. We release the code that allows a user that has a corpus of interest to use $SenDM$ to generate a specific DM list adapted to the corpus, as described above.

4.2   Experimental Setup

The inter-training and evaluation details are identical to those described in Section 3.3. For the number of weakly labeled samples used for inter-training the finance-specific sentiment models, see Table 1 in the appendix. In all cases we divide the samples into training (80%), development (10%), and test (10%) sets.

Finance-specific Corpora   From the General Corpus, C_g, we can define a sub-corpus that is focused on the financial domain, using a provided metadata topic field, and filtering only for articles from the topic 'Finance'. We term this corpus C_{fin} and use it for studying adaptation to the finance domain.

SenDM*.   As we are interested in studying the scenario of adapting to a new domain, which is possibly not covered by the data used to train $SenDM$, and since the corpus used

---

[4]In practical applications, though, users can obviously use our $SenDM$ directly when building a domain-specific model.

[5]In this work we use predictions with $score > 0.9$ and $score < 0.1$ as positive/negative weak labels.
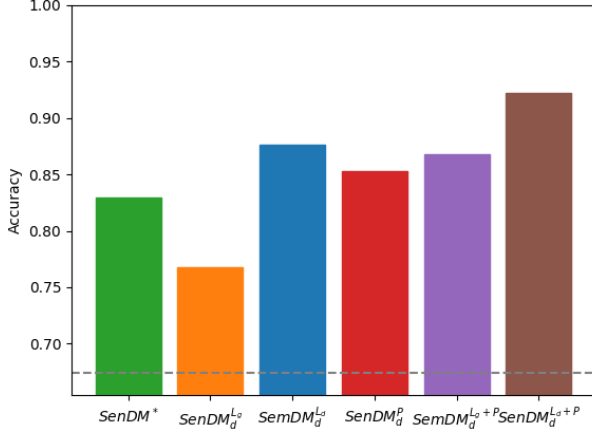
[6]< 0.1 or > 0.9

Figure 3: Performance of the general model and the various domain specific models on the finance dataset, fpb75, for the zero shot setting. ∗: $SenDM^*$ is the general model when trained on a corpus not containing financial documents – see main text for details. All models are base size. In all models the domain d is finance. The dashed horizontal line indicates the prior of the common class in fpb75.

to train $SenDM$ does contain some financial documents, we generate a variant of $SenDM$ excluding the finance domain. This model, denoted by $SenDM^*$, is trained analogously to $SenDM$, except we exclude financial documents from C_g before using it. Naturally, we do not expect a user to train such a model, it is used here and in the appendix, only to examine to what extent our approach can generalize to domains not covered by the general corpus used to train $SenDM$.

4.3 Results

Leveraging $SenDM^*$ and L_{fin}, we generate the five versions of domain specific sentiment models described in 4.1 (see Table 3 for the DMs in L_{fin}[7].). Figure 3 depicts the accuracy of the finance specific sentiment models, on the finance dataset fpb75, for the zero shot setting, in comparison to the accuracy of the general model, $SenDM^*$. As can be seen, indeed using the domain specific DMs rather than the general list improves the accuracy (blue vs. orange, and brown vs. purple bars). One step self-training is valuable even without combining it with a signal from DMs (red vs. green bars). Combining one step self-training in a synergistic manner with the signals from the DMs as described above brings additional value (purple vs. orange and brown vs. blue bars). The highest accuracy is achieved when combining the signal of the finance-specific DMs with one step self training (brown bar). All above accuracy comparisons are significant ($p < 0.05$), based on McNemar's test. It is interesting to note that using L_g for extracting the weak labels from the finance corpus, results in lower performance than using the

---

[7]We also use $SenDM^*$ instead of $SenDM$ to identify L_{fin} for the reasons described above

general model (orange vs. green bars). This may be attributed to a higher noise in the signal of the L_g DMs in the finance corpus compared to the general corpus. This explanation is consistent with the improvement gained by incorporating self-training in a synergistic manner with L_g, a step that results in noise reduction. For the few shot scenario we do not find significant differences between the models – see Figure 2 in the Appendix.

We note that the suggested approach may be applied iteratively to gain further improvement on the finance dataset. Moreover, this adaptation process can also be applied to $SenDM$ itself. We leave these directions for future work.

Analysis of Domain Specific Sentiment DMs As we saw above, leveraging the finance specific DMs, was useful for adapting the sentiment model to the finance domain. Table 3 lists the sentiment-related DMs extracted from general English, L_g, as well as from the financial domain.

Domain	Associated with Positive Sentiment	Associated with Negative Sentiment
General	'fortunately', 'happily', 'hopefully', 'ideally', 'luckily', 'thankfully'	'admittedly', 'curiously', 'inevitably', 'sadly', 'unfortunately'
Finance	'as ORG', 'at the event', 'fortunately', 'hopefully', 'ideally', 'in future', 'in other business', 'luckily', 'once completed','ORG CEO', 'starting DATE', 'thankfully', 'the program', 'this way', 'to achieve this', 'under his leadership', 'with ORG'	according to police', 'sadly', 'the problem', 'the problem is', 'unfortunately', 'worse'
Sports	'beginning DATE', 'fortunately', 'in the future', 'luckily', 'thankfully', 'that way'	'admittedly', 'alas', 'granted', 'ironically', 'sadly', 'true', 'unfortunately', 'unfortunately for ORG'
Science	'established in DATE', 'if necessary', 'if possible', 'if successful', 'luckily', 'that way', 'to address this', 'when possible', 'whenever possible', 'where possible', 'with this approach'	'admittedly', 'at ORDINAL glance', 'at times', 'curiously', 'even then', 'even worse', 'in part', 'inevitably', 'paradoxically', 'predictably', 'regrettably', 'the problem', 'there was', 'too often', 'unsurprisingly', 'without it', 'women'

Table 3: Sentiment-related DMs. The lists below the double line are domain specific. The upper case tokens are NER tags.

Although some of the finance-specific DMs echo those appearing in general English usage (e.g., 'fortunately'), many DMs are domain specific, and in fact may not be considered a DM by the standard linguistic definition. For example, the bigram 'ORG CEO' ('Walmart's CEO', 'BOA's CEO', etc.), is associated with a positive sentiment. This might be surprising at first sight, and would probably not be listed in a manually curated finance-specific DM list, but in hindsight makes sense. When considering what sentences might follow such an opening, one would expect them to discuss positive things about the company. Another such example is 'under the leadership'. Here again, we find, that although not expected *a-priori*, most sentences that follow this opening, would carry a positive sentiment due to the reference to leadership.

Next, we were interested to see how sentiment DMs vary in other domains. We carved out several domain corpora out of the general corpus. Beyond the Finance Corpus described above we also introduce: (1) **The Sports Corpus**: Similarly to the creation of the Finance Corpus but filtering for articles from the topic 'Sports'; and (2) **The Science Corpus**: This corpus focuses on scientific articles from scientific journals, and is defined as any article in C_g that was published in one of the journals included in the list of journal impact factors[8].

These lists can be found in Table 3. We find that some DMs continue to be ubiquitous across domains (e.g., 'fortunately'), but others seem to be associated with sentiment in specific domains only. An interesting example is the word 'women', which in the scientific domain, is among the list of negatively associated DMs. We found that in scientific papers, sentences beginning with 'women' will often deal with issues like oppression and deprivation, and thus are associated with a negative sentiment.

5 Discussion

This work suggests to leverage DMs that carry a sentiment signal to inter-train and adapt general-purpose language models to the sentiment classification task. The obtained sentiment analysis models demonstrate significant performance boost across multiple domains, most notably in the zero-shot and few-shot learning scenarios, emphasizing the practical value of this work. We further show how to evolve the obtained models to a specific domain of interest using automatically identified domain-specific DMs, and show how this approach yields a further performance enhancement in zero-shot learning within the challenging finance domain.

The ability to bootstrap a general, small, and easily identified seed of sentiment carrying DMs into a powerful sentiment analysis model may hold additional valuable implications. For example, this approach can be easily adapted to languages beyond English, including low resource languages, as long as a reasonably sized corpus is available. Another interesting direction would be to expand the proposed approach for *targeted* sentiment analysis. For example, in the finance domain, the company appearing in a sentence can be considered as the sentiment target.

(Sileo et al. 2020) show that various NLP task classes are naturally associated with specific DMs. Thus, the methodology presented here for leveraging DMs to create task-specific language models can be potentially applied to tasks beyond sentiment analysis. Finally, DMs probably represent only one type of linguistic cues among the richness of signals in natural language that can be leveraged as self-supervision to align LMs with downstream tasks.

A Appendix

A.1 Filters Applied to the Domain-Specific DM Selection

In some domains certain DMs may be very specific to a journal, or even a reporter. We find this is the case in the finance domain. Hence, we further filter out DMs whose sentences originate from a relatively narrow set of journals. For this purpose we define the entropy of a DM, based on the probability distribution of its sentences across journals, and filter out 30% DMs with lowest entropy. Furthermore, in the case of the finance domain, since the sentiment analysis task is to identify a sentiment with respect to a company, we apply the above process only to sentences mentioning a company name, where we use the list of all companies traded in one of the five major stock exchanges[9].

A.2 Additional Tables and Figures

Model name	Total number of samples used for intermediate training
$SenDM$	1,876,614
$SenDM^*$	1,815,943
$SenDM_d^{L_g}$	60,671
$SemDM_d^{L_d}$	99,521
$SenDM_d^P$	490,989
$SenDM_d^{L_g+P}$	45,246
$SemDM_d^{L_d+P}$	70,681

Table 4: The number of samples used as weak labels to train each sentiment model. In all cases we divide these into training (80%), development (10%) and test (10%) sets.

References

Araci, D. 2019. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. *ArXiv*, abs/1908.10063.

Bommasani, R.; Hudson, D. A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M. S.; Bohg, J.; Bosselut, A.; Brunskill, E.; Brynjolfsson, E.; Buch, S.; Card, D.; Castellon, R.; Chatterji, N.; Chen, A.; Creel, K.; Davis, J. Q.; Demszky,

---

[8]https://www.scimagojr.com/journalrank.php

[9]Note that this sentence selection step is not applied to the finance corpus which contains all sentences from finance documents
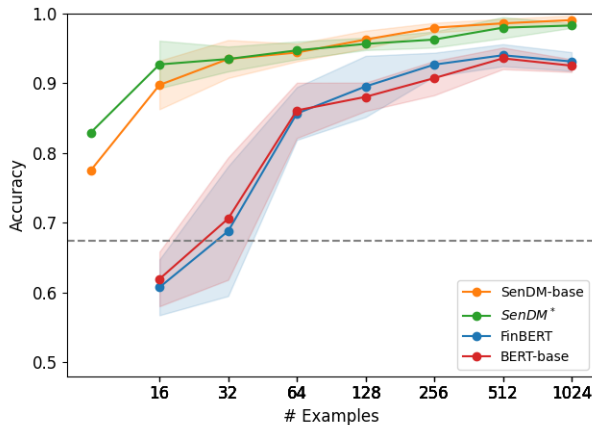
Figure 4: Performance of SenDM-base and base sized baselines on fpb75 given different amounts of fine-tuning examples. Lines indicate the mean and shaded areas indicate the standard deviation over the five seeds (see Section 3.4 for details). The dashed horizontal line indicates the prior of the common class in the dataset. $SenDM^*$ is the same as $SenDM$ when trained excluding financial documents.
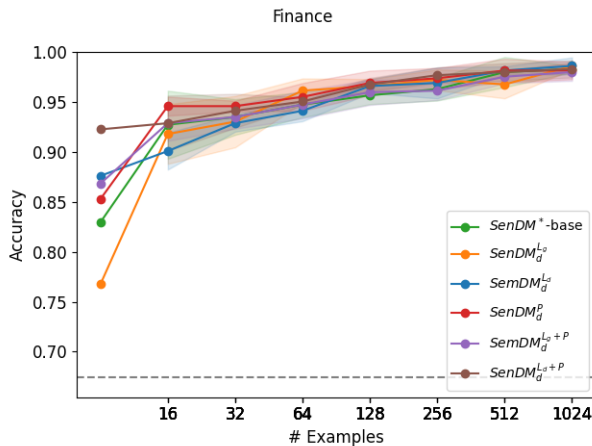

Figure 5: Performance of the general model and the various domain specific models on the finance dataset fpb75 for the zero and few shot settings. $*$: $SenDM^*$ is the general model when trained on the general corpus excluding financial documents – see Section 4.2 for details. In all models the domain d is finance. The dashed horizontal line indicates the prior of the common class in fpb75.

D.; Donahue, C.; Doumbouya, M.; Durmus, E.; Ermon, S.; Etchemendy, J.; Ethayarajh, K.; Fei-Fei, L.; Finn, C.; Gale, T.; Gillespie, L.; Goel, K.; Goodman, N.; Grossman, S.; Guha, N.; Hashimoto, T.; Henderson, P.; Hewitt, J.; Ho, D. E.; Hong, J.; Hsu, K.; Huang, J.; Icard, T.; Jain, S.; Jurafsky, D.; Kalluri, P.; Karamcheti, S.; Keeling, G.; Khani, F.; Khattab, O.; Kohd, P. W.; Krass, M.; Krishna, R.; Kuditipudi, R.; Kumar, A.; Ladhak, F.; Lee, M.; Lee, T.; Leskovec, J.; Levent, I.; Li, X. L.; Li, X.; Ma, T.; Malik, A.; Manning, C. D.; Mirchandani, S.; Mitchell, E.; Munyikwa, Z.; Nair, S.; Narayan, A.; Narayanan, D.; Newman, B.; Nie, A.; Niebles, J. C.; Nilforoshan, H.; Nyarko, J.; Ogut, G.; Orr, L.; Papadimitriou, I.; Park, J. S.; Piech, C.; Portelance, E.; Potts, C.; Raghunathan, A.; Reich, R.; Ren, H.; Rong, F.; Roohani, Y.; Ruiz, C.; Ryan, J.; Ré, C.; Sadigh, D.; Sagawa, S.; Santhanam, K.; Shih, A.; Srinivasan, K.; Tamkin, A.; Taori, R.; Thomas, A. W.; Tramèr, F.; Wang, R. E.; Wang, W.; Wu, B.; Wu, J.; Wu, Y.; Xie, S. M.; Yasunaga, M.; You, J.; Zaharia, M.; Zhang, M.; Zhang, T.; Zhang, X.; Zhang, Y.; Zheng, L.; Zhou, K.; and Liang, P. 2021. On the Opportunities and Risks of Foundation Models. arXiv:2108.07258.

Braud, C.; and Denis, P. 2016. Learning Connective-based Word Representations for Implicit Discourse Relation Identification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 203–213. Austin, Texas: Association for Computational Linguistics.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Chalkidis, I.; Fergadiotis, M.; Malakasiotis, P.; Aletras, N.; and Androutsopoulos, I. 2020. LEGAL-BERT: The Muppets straight out of Law School. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2898–2904. Online: Association for Computational Linguistics.

Chang, W.-C.; Yu, F. X.; Chang, Y.-W.; Yang, Y.; and Kumar, S. 2020. Pre-training tasks for embedding-based large-scale retrieval. *arXiv preprint arXiv:2002.03932*.

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.

Grave, E.; Bojanowski, P.; Gupta, P.; Joulin, A.; and Mikolov, T. 2018. Learning Word Vectors for 157 Languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA).

Gu, Y.; Zhang, Z.; Wang, X.; Liu, Z.; and Sun, M. 2020. Train no evil: Selective masking for task-guided pre-training. *arXiv preprint arXiv:2004.09733*.

Gururangan, S.; Marasović, A.; Swayamdipta, S.; Lo, K.; Beltagy, I.; Downey, D.; and Smith, N. A. 2020. Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Huber, P.; Aghajanyan, A.; Oğuz, B.; Okhonko, D.; tau Yih, W.; Gupta, S.; and Chen, X. 2021. CCQA: A New Web-Scale Question Answering Dataset for Model Pre-Training. arXiv:2110.07731.

Jernite, Y.; Bowman, S. R.; and Sontag, D. 2017. Discourse-Based Objectives for Fast Unsupervised Sentence Representation Learning. arXiv:1705.00557.

Jiao, X.; Yin, Y.; Shang, L.; Jiang, X.; Chen, X.; Li, L.; Wang, F.; and Liu, Q. 2020. TinyBERT: Distilling BERT for Natural Language Understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4163–4174. Online: Association for Computational Linguistics.

Joshi, M.; Chen, D.; Liu, Y.; Weld, D. S.; Zettlemoyer, L.; and Levy, O. 2019. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *CoRR*, abs/1907.10529.

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.

Keung, P.; Lu, Y.; Szarvas, G.; and Smith, N. A. 2020. The Multilingual Amazon Reviews Corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

LeCompte, T.; and Chen, J. 2017. Sentiment Analysis of Tweets Including Emoji Data. In *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*, 793–798.

Levy, R.; Bogin, B.; Gretz, S.; Aharonov, R.; and Slonim, N. 2018. Towards an argumentative content search engine using weak supervision. In *Proceedings of the 27th International Conference on Computational Linguistics*, 2066–2081.

Liu, Y.; and Li, S. 2016. Recognizing Implicit Discourse Relations via Repeated Reading: Neural Networks with Multi-Level Attention. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1224–1233. Austin, Texas: Association for Computational Linguistics.

Malo, P.; Sinha, A.; Korhonen, P.; Wallenius, J.; and Takala, P. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65.

Meng, Y.; Zhang, Y.; Huang, J.; Xiong, C.; Ji, H.; Zhang, C.; and Han, J. 2020. Text classification using label names only: A language model self-training approach. *arXiv preprint arXiv:2010.07245*.

Moder, C. L.; and Martinovic-Zic, A. 2004. *Discourse across languages and cultures*, volume 68. John Benjamins Publishing.

Nie, A.; Bennett, E.; and Goodman, N. 2019. DisSent: Learning Sentence Representations from Explicit Discourse Relations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4497–4510. Florence, Italy: Association for Computational Linguistics.

Pan, B.; Yang, Y.; Zhao, Z.; Zhuang, Y.; Cai, D.; and He, X. 2018. Discourse Marker Augmented Network with Reinforcement Learning for Natural Language Inference. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 989–999. Melbourne, Australia: Association for Computational Linguistics.

Pruksachatkun, Y.; Phang, J.; Liu, H.; Htut, P. M.; Zhang, X.; Pang, R. Y.; Vania, C.; Kann, K.; and Bowman, S. R. 2020. Intermediate-task transfer learning with pretrained models for natural language understanding: When and why does it work? *arXiv preprint arXiv:2005.00628*.

Rutherford, A.; and Xue, N. 2015. Improving the inference of implicit discourse relations via classifying explicit discourse connectives. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 799–808.

Shnarch, E.; Gera, A.; Halfon, A.; Dankin, L.; Choshen, L.; Aharonov, R.; and Slonim, N. 2021. Cluster & Tune: Enhance {BERT} Performance in Low Resource Text Classification.

Sileo, D.; de Cruys, T. V.; Pradel, C.; and Muller, P. 2020. DiscSense: Automated Semantic Analysis of Discourse Markers. arXiv:2006.01603.

Sileo, D.; Van De Cruys, T.; Pradel, C.; and Muller, P. 2019. Mining Discourse Markers for Unsupervised Sentence Representation Learning. 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)*, 3477–3486. Minneapolis, Minnesota: Association for Computational Linguistics.

Sun, Y.; Wang, S.; Li, Y.; Feng, S.; Chen, X.; Zhang, H.; Tian, X.; Zhu, D.; Tian, H.; and Wu, H. 2019. Ernie: Enhanced representation through knowledge integration. *arXiv preprint arXiv:1904.09223*.

Tian, H.; Gao, C.; Xiao, X.; Liu, H.; He, B.; Wu, H.; Wang, H.; and Wu, F. 2020. SKEP: Sentiment knowledge enhanced pre-training for sentiment analysis. *arXiv preprint arXiv:2005.05635*.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. 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*, 353–355. Brussels, Belgium: Association for Computational Linguistics.

Wu, C.-S.; Hoi, S. C.; Socher, R.; and Xiong, C. 2020. TOD-BERT: Pre-trained Natural Language Understanding for Task-Oriented Dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 917–929. Online: Association for Computational Linguistics.

Zhang, J.; Zhao, Y.; Saleh, M.; and Liu, P. 2020a. PEGA-SUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 11328–11339. PMLR.

Zhang, J.; Zhao, Y.; Saleh, M.; and Liu, P. 2020b. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, 11328–11339. PMLR.

Zhang, T.; Wu, F.; Katiyar, A.; Weinberger, K. Q.; and Artzi, Y. 2020c. Revisiting few-sample BERT fine-tuning. *arXiv preprint arXiv:2006.05987*.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-Level Convolutional Networks for Text Classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, 649–657. Cambridge, MA, USA: MIT Press.

Zhou, J.; Tian, J.; Wang, R.; Wu, Y.; Xiao, W.; and He, L. 2020. SentiX: A Sentiment-Aware Pre-Trained Model for Cross-Domain Sentiment Analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, 568–579. Barcelona, Spain (Online): International Committee on Computational Linguistics.