Domain-agnostic Document Representation Learning Using Latent Topics and Metadata

Natraj Raman[1], **Armineh Nourbakhsh**[2], **Sameena Shah**[2], **Manuela Veloso**[2]
J.P.Morgan AI Research
[1]London, UK.
[2]New York, USA.
first.last@jpmorgan.com

Abstract

Fine-tuning a pre-trained neural language model with a task specific output layer is the de facto approach of late when dealing with document classification. This technique is inadequate when labeled examples are unavailable at training time and when the metadata artifacts in a document must be exploited. We address these challenges by generating document representations that capture both text and metadata in a task agnostic manner. Instead of traditional auto-regressive or auto-encoding based training, our novel self-supervised approach learns a soft-partition of the input space when generating text embeddings by employing a pre-learned topic model distribution as surrogate labels. Our solution also incorporates metadata explicitly rather than just augmenting them with text. The generated document embeddings exhibit compositional characteristics and are directly used by downstream classification tasks to create decision boundaries from a small number of labels, thereby eschewing complicated recognition methods. We demonstrate through extensive evaluation that our proposed cross-model fusion solution outperforms several competitive baselines on multiple domains.

Introduction

The current popularity of transformer (Vaswani et al. 2017) based models in NLP is owed to their capacity in constructing semantically rich representations and to their ability to accommodate transfer-learning. This enables users to pre-train language models on large unlabeled corpora, and then fine-tune the representations using smaller sets of labeled examples (Howard and Ruder 2018). However, the state-of-the-art research for few-short learning is still far from human performance on many tasks. This is even more pronounced in zero-shot experiments, where no labeled data is available at training time.

Due to the emphasis on scalability and generalizability of these models, what is often left out of consideration is the practical aspects of how these models are commonly applied in real-world settings, where the datasets are accompanied by metadata tags that include some signal about the nature and content of each document in the corpus. This metadata is often stripped before models are applied to the text, in order to avoid convoluted and bespoke architectures. In some cases, the metadata is simply concatenated to the document (Zhang et al. 2020) without specific controls on how representations are generated from raw text versus metadata tags.

Another aspect that current research often leaves out is the topic distribution of the unlabeled dataset. Transformer models are commonly trained in a stochastic fashion, where the global composition of topics in the corpus is not explicitly built into the loss function. Often the pre-training corpus is large enough for this effect to likely be negligible, but when the corpus is smaller, this lack of insight into global distributional statistics may have an impact on the compositionality of the resulting representations.

In this study, we explore how addressing these issues can improve the performance of transformer-based models on document classification tasks. We propose a framework that encapsulates universal distributional statistics about the raw text, as well as encodings for metadata tags. Figure 1 illustrates our framework when applied to a hypothetical dataset of product reviews with a diverse set of metadata artifacts. The raw text of each review is paired with other metadata artifacts available, such as user profile information, product identifiers, and location information. All the artifacts are fed into a deep learning model with the self-learning mechanism adjusted to match the type of artifact presented. For example, for the raw text of the reviews, instead of using a standard Masked Language Model (MLM) objective (Devlin et al. 2018), the model learns by predicting the latent topic distribution of the text based on a pre-trained generative model such as LDA (Blei, Ng, and Jordan 2003). For categorical metadata such as product identifiers and location, the model predicts the specific metadata tag. Certain metadata can also be directly encoded, bypassing the self-learning task. The resulting representations are semantically rich and can be plugged into a simple K-NN model for various label-prediction tasks, bypassing the need for complicated, task-specific classification models.

We demonstrate how the representations created by our framework exhibit compositional characteristics that can be useful to granular classification tasks. While small and scalable to many different settings, our framework improves the performance of transformer-based language models on classification tasks on a variety of datasets from different do-

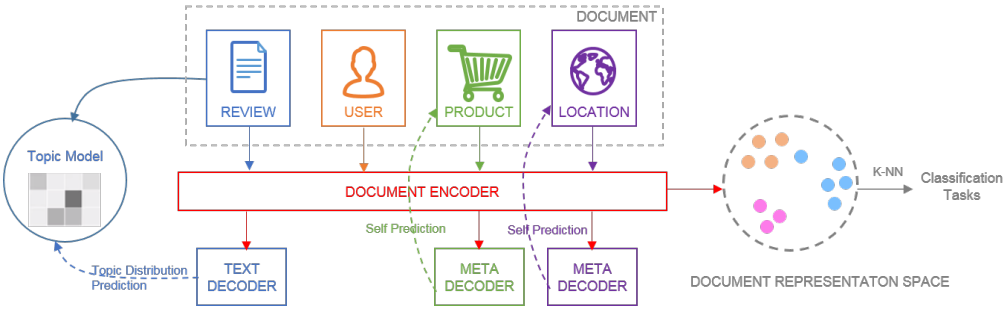

Figure 1: Multi-part input document is converted into an embedding representation. Self-supervision is based on latent topic distribution and (optional) metadata reconstruction. An input is classified based on its neighborhood in the representation space.

mains. Our experiments show that regardless of the underlying neural architecture, performance is enhanced by a robust minimum of 5% over a conventional fine-tuned model. The remaining sections of this paper lay out our methodology, describe our datasets, and present experimental results.

Related Work

Semi-supervised learning in neural language models has largely focused on "pre-train and fine-tune" pipelines (Howard and Ruder 2018), which take advantage of large unlabeled datasets, paired with small labeled datasets. Researchers have explored few-sample learning (Sharaf, Hassan, and au2 2020), to handle prohibitively small labeled datasets. However, performance on NLP tasks remains far from human baselines (Wang et al. 2020) due to the fact that the inductive fine-tuning process fails to directly take advantage of a dataset's universal distributional characteristics beyond what is encoded in the pre-trained representations. These representations are sometimes paired with other distributional signal such as topic models and tf-idf vectors (Lim and Madabushi 2020).

Transductive learning approaches actively take advantage of sample distributions during inference. This paradigm has been used to improve performance on tasks such as cross-domain text classification (Ionescu and Butnaru 2018) and neural machine translation (Poncelas, de Buy Wenniger, and Way 2019). However, these studies do not address cases where the original dataset lacks enough labeled examples. Similarly, multi-task learning studies have addressed cases where the model can be robustly trained for one task such as entity extraction, and scale to other tasks such as co-reference resolution (Sanh, Wolf, and Ruder 2018).

In this study, we propose a transductive framework that can take advantage of a limited labeled dataset paired with a larger unlabeled dataset to generate rich representations for document classification tasks.

Model

We first introduce the encoder architecture that is used to obtain an input representation that captures both the text and metadata information in a task agnostic manner. Then we present the decoder structure that employs self-supervision to define the training objective. Finally, we discuss how the learned representation is directly used in downstream classification tasks. Figure 2 provides an overview of our model.

Representation Learning

Given N training examples $X = \{x^1, ..., x^N\}$, let $x = (\tau, m_1, ..., m_P)$ be an input that contains text τ accompanied with P different metadata artifacts m. The text consist of T tokens $(\tau_1, ..., \tau_T)$ and each metadata m_p is a sequence $(m_{p1}, ..., m_{pl}, ...m_{pL})$ of fixed length L. Sequences shorter than T or L are simply padded. Let $m_{pl} \in \Omega^p$, where Ω^p is the discrete set of information for p^{th} metadata.

The text input is converted into an interim embedding representation ϕ using a function

$$\mathbf{f} : \tau \rightarrow \phi, \qquad \phi \in \mathbb{R}^{D_t} \tag{1}$$

where D_t is the text embedding size. The function $\mathbf{f}$ is a transformer model that employs self-attention mechanism to capture dependencies between arbitrary positions of text in an efficient manner. The input tokens are augmented with a special token $[CLS]$ that represents the aggregate information of the entire sequence. The output corresponding to this special token at the last layer is used as ϕ.

There are P independent non-linear functions to convert each metadata input into an interim embedding ψ:

$$\mathbf{g}_p : m_{pl} \rightarrow \psi_{pl}, \qquad \psi_{pl} \in \mathbb{R}^{D_p}, \quad \forall p = 1...P, l = 1...L \tag{2}$$

where D_p is the metadata embedding size. We make use of a feed-forward network with multiple layers as the conversion function $\mathbf{g}$, with each layer comprising of a linear transformation followed by a non-linear activation. If a metadata cannot be meaningfully interpreted (e.g. product code or user id), we use one-hot encoding of the metadata values as input. Otherwise, the input is set to a 300 dimensional vector derived by averaging the Glove (Pennington, Socher, and Manning 2014) vectors corresponding to the words in metadata text. The embedding for a metadata sequence is aggregated using a mean function that masks out padded positions $\gamma \in \{0, 1\}$ as

$$\psi_p = \frac{1}{L} \sum_l \gamma_{pl} \psi_{pl}. \tag{3}$$

The final embedding representation for an input x is obtained by first concatenating the text and metadata embed-

dings and then projecting them to a lower-dimensional space using a linear transformation as follows:

$$z = W_z^\top (\phi \oplus \psi_1 \oplus ... \oplus \psi_P), \qquad z \in \mathbb{R}^{D_e} \tag{4}$$

where z is the final input embedding of size D_e and $W_z \in \mathbb{R}^{(D_t + \sum_p D_p) \times D_e}$ is a parameter matrix.

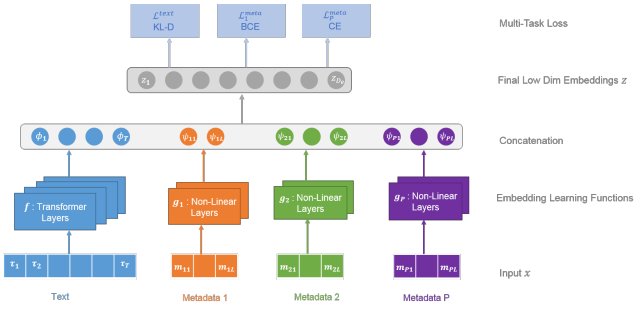

Figure 2: Model Architecture. Embeddings learned independently for different input types are combined and then projected to a lower dimensional space. The model is trained using a multi-task objective function.

Self-supervised Loss

The key idea in self-supervision is to generate synthetic labels automatically from the data and use these labels to construct loss functions. Instead of the typical word masking solution, we propose a cross-model fusion approach. A topic model for the text corpora is learned in an unsupervised manner and the inferred topic distribution is used as synthetic labels. By discovering latent semantics embedded in the text, topic models introduce a partition of the input space. Importantly, the mixed membership of topics enables a soft partition rather than a hard partition, thereby efficiently handling overlapping boundaries and complex structures.

Formally, the topic distribution φ corresponding to an input text is obtained using a function

$$\mathbf{h} : \tau \rightarrow \varphi, \qquad \varphi \in \mathbb{R}^K \tag{5}$$

where K is the number of topics and $\mathbf{h}$ is a topic modeling function based on LDA. The input embeddings obtained using (4) are first linearly projected into the topic space as:

$$\lambda^n = W_t^\top z^n, \qquad \lambda^n \in \mathbb{R}^K. \tag{6}$$

Here λ^n is the projected distribution in topic space for the n^{th} training example and $W_t \in \mathbb{R}^{D_e \times K}$ is a parameter matrix. The Kullback-Leibler (KL) divergence between the prelearned topic distribution and the input projection is minimized during training. This translates into the following loss function for text inputs:

$$\mathcal{L}^{text} = \sum_n \sum_k \varphi_k^n \, log \, \frac{\varphi_k^n}{\lambda_k^n}. \tag{7}$$

For the metadata, we employ a loss function that aims to minimize the reconstruction error between the input metadata value and the value decoded from the final embedding representation. Given that each metadata is a sequence of values from a discrete set, a multi-label binary cross-entropy loss is an appropriate choice.

Let there be V^p possible values for the p^{th} metadata i.e. $|\Omega^p| = V^p$ and let $y_p \in \{0, 1\}^{V^p}$ denote the multi-label values consolidated from an input metadata sequence. A linear transformation decoder layer first converts the input embeddings into the metadata space as

$$\zeta_p^n = W_p^\top z^n, \qquad \zeta_p^n \in \mathbb{R}^{V^p} \tag{8}$$

where ζ_p^n is the projection for the n^{th} input and $W_p \in \mathbb{R}^{D_e \times V^p}$ is a parameter matrix as before. The reconstruction loss function is formulated as:

$$\mathcal{L}_p^{meta} = \sum_n \sum_v -y_{p,v} \, log \, \sigma(\zeta_{p,v}^n) - (1 - y_{p,v}) \, log \, (1 - \sigma(\zeta_{p,v}^n)) \tag{9}$$

where σ is the standard sigmoid function.

Using the text and metadata losses in (7) and (9), the training objective is framed as

$$\min_\theta \; \omega^{text} \mathcal{L}^{text} + \sum_p \omega_p^{meta} \mathcal{L}_p^{meta} \tag{10}$$

where θ is the set of all model parameters and ω is a real-valued hyper-parameter that controls the relative importance between the text and various metadata.

Unseen Classification Tasks

The input representations obtained using the above model have several desirable characteristics: the embeddings are in a compact form because of the projection into a lower-dimensional space, the salient information in the inputs are preserved by the reconstruction loss, and similar points are grouped together with the use of soft-partition labels. Hence these embeddings can be used as-is in a non-parametric setting for downstream classification tasks. This approach is particularly attractive in situations where it is expensive to train new classification models or it may not be possible to perform training due to the scarcity of labeled examples.

Given a small number of labeled examples, we use nearest neighbor technique to compute the classification label of a query point. Specifically, the Euclidean distance between the embeddings of the query point and the labeled points is used to identify the nearest neighbours. The query point's class label is computed using a mode function on the nearest neighbour labels.

Experiments

Experiments are conducted on three real-world datasets that vary in text style, application domain and nature of metadata. We demonstrate using these datasets that the proposed solution outperforms standard self-supervised training objectives for a variety of benchmark models.

Setup

We use three datasets in our experiments: *Github-AI* (Zhang et al. 2019) dataset that contains a list of source code repositories, *Amazon* (McAuley and Leskovec 2013) dataset that

Table 1: Samples form each of the three datasets.

Github-AI	Description	Handbag GAN. In this project I will implement a DCGAN to see if I can generate handbag designs...
	Repo Name	GAN experiments
	Tags	gan,dcgan,deep-learning,google-cloud
	Labels	Image Generation (granular), Computer Vision (coarse)
Amazon	Review	Best little ice cream maker works well, not too noisy, easy to clean. Recommend buying extra...
	Product	B00000JGRT
	Label	Home_and_Kitchen
Twitter	Tweet	greek yogurt fresh fruit honey granola healthy living bakery.
	Hashtags	#healthyliving, #bakeri
	Label	Food

collects review data from online retailer Amazon and *Twitter* (Zhang et al. 2017) dataset with tweets collected during a three month period are used for evaluation. All these datasets are publicly available and Table 1 lists a few samples.

For comparison purposes, we use five state-of-the-art neural language models namely *BERT* (Devlin et al. 2018), *DistilBERT* (Sanh et al. 2019), *XLNet* (Yang et al. 2019), *RoBERTa* (Liu et al. 2019) and *Electra* (Clark et al. 2020). All these models are based on the transformer architecture with variations in their training procedure.

There are three different evaluation setups corresponding to these models. In the *No Finetuning* setup, the side information such as repository name or hashtag is augmented with text data to create a single text block. The *LM Finetuning* setup has the language model finetuned on the augmented text blocks before performing inference. The *Our Approach* setup reflects the architecture described above.

We employ the base configuration of a transformer model, which typically has 12 layers and 768 hidden neurons per token. All hyper-parameters were chosen after a careful grid search and these values include an initial learning rate of $5e-5$, a drop out probability of 0.1, sequence length of 512 and batch size of 8. The metadata embedding size D_p is set to 50 while the final input embedding size D_e is set to 500. Only 10% of the data is used as exemplars for nearest neighbour classification with all points in the neighbourhood weighted equally.

Results

The F1 scores of the classification results are shown in Table 2. We see that our approach of using a loss function based on topic distribution significantly improves the classifier performance for all the three datasets across all the five benchmark models. DistilBERT trained using our objective function has a best F1 Score of 50.0% for Github-AI and 50.7% for Twitter respectively. These two datasets are particularly challenging due to the large number of classes and a small sample size. In contrast to the above, the Amazon dataset has a large training set size and finetuning the language model results in markedly better classifier performance. However, our proposed training objective still outperforms all the other baselines for this dataset with the XLNet model having the best F1 Score of 88.8%.

Discussion

Our self-supervision method aligns the input embeddings to reflect topic distributions, thereby inducing a soft clustering of input points with those points that share similar characteristics appearing together. The embeddings generated by standard language models do not necessarily have this clustering property. Figure 3 makes this partitioning effect evident. It plots in 2-D the inferred input embeddings for Amazon dataset. The top section of this figure contains the embeddings from a finetuned language model and there are no discernible clusters here. However, in the bottom section that corresponds to the same model trained using our objective function, we can clearly see patterns of points appearing together. This natural grouping of the inputs enable identification of class labels based on neighbourhood search very effective when compared with standard masked language modeling.

The learned embeddings also capture topic characteristics with input points corresponding to the same latent topic placed together. To validate this, we over-cluster the embeddings using KMeans and qualitatively examine the cluster contents. The left side of Figure 4 contains one such cluster, where the discussions around *locks* in the Amazon review are consolidated into the same cluster. Furthermore, semantically similar labels appear near to each other in the embedding space. We observe that the cluster closest to an *Apps_for_Android* cluster is the *Video_Games* cluster. Similarly, a *CDs_and_Vinyl* cluster is close to *Movies_and_TV* cluster as illustrated in the right side of Figure 4. This level of compositionality opens the model up for applications to hierarchical classification and clustering.

We explore the robustness of the framework through ablation studies. Figure 5 plots the effect of different topic size on the classifier performance for GitHub-AI dataset. Having extremely few topics does affect the model performance. However, the results are stable for a wide range of topic sizes. Hence the model is not sensitive to an exact value of the topic size hyper-parameter. We also ask how the baseline model performs in the presence of large amounts of training data. Figure 6 compares the performance between our model and *LM Finetuning* setup for the Amazon dataset. We see that our model performs significantly better when there is fewer training data available, with finetuned model converging when training size increases beyond a threshold.

Table 2: Comparison of Classification Results - F1 Scores

Dataset	Transformer	No Finetuning	LM Finetuning	Our Solution
Github-AI	BERT	22.8	27.2	45.0
	DistilBERT	26.4	28.4	**50.0**
	XLNet	21.6	19.2	46.5
	RoBERTa	21.3	19.9	34.4
	Electra	21.3	20.8	29.5
Amazon	BERT	32.0	78.1	86.5
	DistilBERT	54.3	84.6	88.6
	XLNet	19.0	28.1	**88.8**
	RoBERTa	20.8	73.3	88.2
	Electra	22.0	37.9	86.4
Twitter	BERT	22.2	40.1	46.0
	DistilBERT	32.4	44.3	**50.7**
	XLNet	18.0	26.9	40.9
	RoBERTa	15.0	29.3	22.7
	Electra	21.0	15.3	42.9

Figure 3: Embedding Visualization. *top*: Embeddings produced using standard language model training objective. *bottom*: Embeddings produced with loss function based on topic distribution with a perceptible grouping of points from the same class.

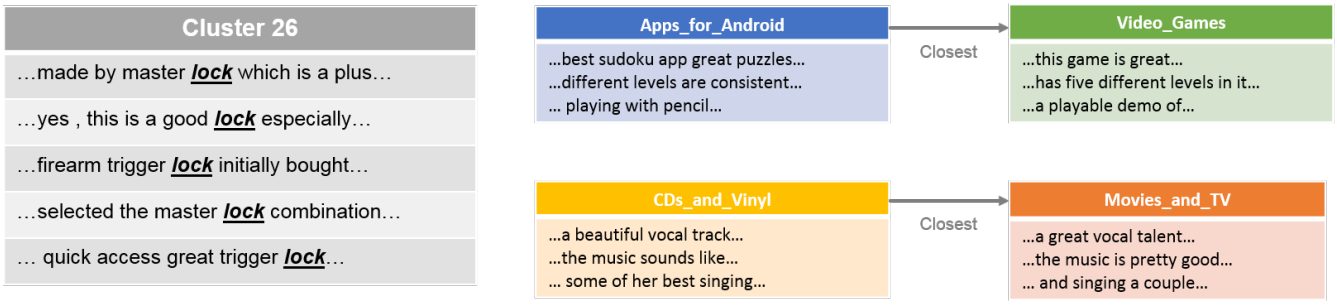

Figure 4: Semantic composition in learned embeddings. *left*: Discussions around *lock* topic occur in the same cluster. *right*: Semantically similar labels appear close together in the embedding space.

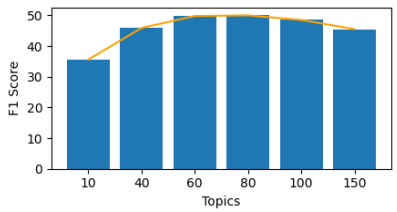

Figure 5: Topic size effect on classifier performance.

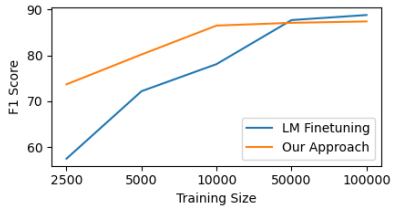

Figure 6: Training set size impact on classifier performance.

Conclusion

In this paper, we presented a flexible framework that combines latent topic information and metadata encodings with transformer-based models to learn semantically rich document representations that can be used for classification tasks in a transductive fashion. We show 4%+ improvement over out-of-the-box pre-trained embeddings as well as conventional fine-tuning on a variety of datasets. We also qualitatively illustrate the semantic compositionality of the resulting embeddings. Our framework is especially effective when training data is smaller or when metadata tags provide useful semantic signal that would otherwise be missed. In future work, we plan to explore the effectiveness of our framework in unsupervised hierarchical clustering.

Acknowledgments

This paper was prepared for information purposes by the Artificial Intelligence Research group of JPMorgan Chase & Co and its affiliates ("JP Morgan"), and is not a product of the Research Department of JP Morgan. J.P. Morgan makes no representation and warranty whatsoever and disclaims all liability for the completeness, accuracy or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

References

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3(Jan):993–1022.

Clark, K.; Luong, M.-T.; Le, Q. V.; and Manning, C. D. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Howard, J., and Ruder, S. 2018. Universal language model fine-tuning for text classification.

Ionescu, R. T., and Butnaru, A. M. 2018. Transductive learning with string kernels for cross-domain text classification.

Lim, W. M., and Madabushi, H. T. 2020. Uob at semeval-2020 task 12: Boosting bert with corpus level information.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

McAuley, J., and Leskovec, J. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, 165–172.

Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global vectors for word representation. In *Proceedings of EMNLP*, 1532–1543.

Poncelas, A.; de Buy Wenniger, G. M.; and Way, A. 2019. Transductive data-selection algorithms for fine-tuning neural machine translation.

Sanh, V.; Debut, L.; Chaumond, J.; and Wolf, T. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Sanh, V.; Wolf, T.; and Ruder, S. 2018. A hierarchical multi-task approach for learning embeddings from semantic tasks.

Sharaf, A.; Hassan, H.; and au2, H. D. I. 2020. Meta-learning for few-shot nmt adaptation.

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

Wang, Y.; Yao, Q.; Kwok, J.; and Ni, L. M. 2020. Generalizing from a few examples: A survey on few-shot learning.

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R. R.; and Le, Q. V. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, 5754–5764.

Zhang, C.; Zhang, K.; Yuan, Q.; Tao, F.; Zhang, L.; Hanratty, T.; and Han, J. 2017. React: Online multimodal embedding for recency-aware spatiotemporal activity modeling. In *Research and Development in Information Retrieval*, 245–254.

Zhang, Y.; Xu, F. F.; Li, S.; Meng, Y.; Wang, X.; Li, Q.; and Han, J. 2019. Higitclass: Keyword-driven hierarchical classification of github repositories. In *2019 IEEE International Conference on Data Mining (ICDM)*, 876–885. IEEE.

Zhang, Y.; Meng, Y.; Huang, J.; Xu, F. F.; Wang, X.; and Han, J. 2020. Minimally supervised categorization of text with metadata.