Using Transformer-based Models for Taxonomy Enrichment and Sentence Classification

Parag Dakle, Shrikumar Patil, SaiKrishna Rallabandi, Chaitra Hegde, Preethi Raghavan
Fidelity Investments, AICoE, Boston
{paragpravin.dakle, shrikumarrajendra.patil, saikrishna.rallabandi, chaitra.vishwanathahegde, preethi.raghavan}@fmr.com

Abstract

In this paper, we present a system that addresses the taxonomy enrichment problem for "Environment, Social and Governance" issues in the financial domain, as well as classifying sentences as sustainable or unsustainable, for FinSim4-ESG, a shared task for the FinNLP workshop at IJCAI-2022. We first created a derived dataset for taxonomy enrichment by using a sentence-BERT-based paraphrase detector (Reimers and Gurevych, 2019) (on the train set) to create positive and negative term-concept pairs. We then model the problem by fine-tuning the sentence-BERT-based paraphrase detector on this derived dataset, and use it as the encoder, and use a Logistic Regression classifier as the decoder, resulting in test Accuracy: 0.6 and Avg. Rank: 1.97. In case of the sentence classification task, the best performing classifier (Accuracy: 0.92) consists of a pre-trained RoBERTa model (Liu et al., 2019a) as the encoder and a Feed Forward Neural Network classifier as the decoder.

1 Introduction

Taxonomies classify, categorize and organize information hierarchically and are typically designed and curated by domain experts. They require frequent manual and automated updates to capture a domain sufficiently and to be considered complete. However, it is not to feasible to manually edit taxonomies to reflect changing concepts and evolving human knowledge. The taxonomy enrichment task helps address this problem by developing methods to add new terms to an existing taxonomy. The FinNLP shared task 1 defines this problem on a ESG taxonomy. Given a list of concepts and terms, the task is to rank the concepts given the term. In case of shared task 2, we are asked to classify a given sentence from sustainability reports and other documents as either sustainable or unsustainable.

In approaching these problems, we leverage large-scale pre-trained language models for token and sentence representations. We explore transfer learning through transformer models like BeRT (Devlin et al., 2018), DistillBeRT (Sanh et al., 2019), RoBeRTa (Liu et al., 2019b) as well as generative text to text transformers like T5 (Raffel et al., 2019) especially since training data is very limited for both tasks.

Like most NLP tasks in FinTech, the task 1 has limited amount of data. We addressed this limitation by creating a dataset derived from the train set and used a paraphrase detector to create positive and negative instances of <term, concept> pairs. We then fine-tune sentence-BERT (Reimers and Gurevych, 2019) on this derived dataset and use it as the encoder in our model. The decoder is a logistic regression classifier. This gives us a ten-fold cross-validated accuracy of 0.89 on the train set. However at test time, the performance varies and resulting accuracy is 60.6%. We describe the different approaches to modeling this problem that led to this final system and hypothesize reasons for the train-test performance discrepancy in the final system.

Shared task 2 is a binary sustainability classification task. We experimented with various models starting with a tf-idf based classifier to transformer based RoBeRTa (Liu et al., 2019b) based classifier. The RoBeRTa based model resulted in a ten-fold cross-validated accuracy of 0.96 and test-set accuracy of 0.92.

2 Related Works

2.1 Taxonomy Enrichment

Taxonomy enrichment is the task of extending an existing taxonomy with new terms. Word embeddings derived from language models are popularly used for this task (Jurgens and Pilehvar, 2016; Nikishina et al., 2021). Using word vector representations, it may be modeled as a hypernym classification task (SemEval 2018) or an embedding

Proceedings of the Fourth Workshop on Financial Technology and Natural Language Processing (FinNLP), pages 250 - 258

similarity task. Graph based representations are also used for taxonomy completion tasks (Zeng et al., 2021).

We explore the taxonomy enrichment problem using embedding similarity by modeling the problem as a paraphrase detection task. In the taxonomy enrichment task, we are given a list of terms and corresponding concepts. Our approach uses word2vec to get sentence embeddings for terms; we use (Reimers and Gurevych, 2019) which learns semantic representation of the given sentence using contrastive loss trained on various open-source datasets (Bowman et al., 2015; Williams et al., 2018).

2.2 Sustainability Classification

Pre-trained language models such as BERT(Devlin et al., 2018) and Roberta(Liu et al., 2019b) have achieved state-of-the-art performance on classification tasks. In our experiments, we found that Roberta (Liu et al., 2019a) performs better than other models.

3 Problem Statement

3.1 Sub-task 1: Taxonomy Enrichment

Given a set T of n terms $\{t_1, t_2, .., t_n\}$ and a set C of m concepts $\{c_1, c_2, .., c_m\}$, the task of taxonomy enrichment is to find a many-to-one mapping M between the terms and the corresponding concepts.

3.2 Sub-task 2: Sentence Classification

Given a set of k sentences $S = \{s_1, s_2, .., s_k\}$, the aim of this sub-task is to classify each sentence in S into one of two classes - *sustainable* or *unsustainable*.

4 Data Description

The training dataset for sub-task 1 contains 646 annotated term-concept pairs. The total number of unique concepts are 25. Table 1 shows the label distribution in the training set for sub-task 1. Since the released training data did not contain any validation set, 10-fold cross validation was used for training. The data was first shuffled and then split into 10 parts. For each fold, 9 parts containing 582 term-concept pairs and one fold containing 65 term-concept pairs were selected as the training and validation set respectively.

The training dataset for sub-task 2 contains 2265 annotated sentences. Table 2 shows the label distribution in the training set for sub-task 2. On an average a sentence in the training set had a length of 162 characters or 25 tokens. Similar to sub-task 1, for this sub-task also 10-fold cross validation was used. Each fold contains 2038 sentences in the training set and 227 sentences in the validation set. In addition to the training sets for both sub-tasks, the shared task also provided a set of 190 annual reports and sustainability reports of financial companies.

Concept	#instances
Energy efficiency and renewable energy	59
Sustainable Food & Agriculture	54
Product Responsibility	51
circular economy	47
Sustainable Transport	46
Emissions	39
Shareholder rights	38
Board Make-Up	37
Injury frequency rate for subcontracted labour	35
Executive compensation	32
Biodiversity	29
Community	27
Employee engagement	23
Employee development	22
Water & waste-water management	21
Carbon factor	19
Future of work	18
Waste management	16
Recruiting and retaining employees (incl. work-life balance)	11
Human Rights	10
Audit Oversight	7
Injury frequency rate	2
Board Independence	2
SHARE CAPITAL	2
Total	**646**

Table 1: Label distribution in the training set for taxonomy enrichment sub-task 01

5 Taxonomy Enrichment Task

5.1 Preliminary Experiments and Results

- Baseline 1 (B_1): A Word2Vec model trained on the given reports is used to generate term and concept embeddings. The similarity scores or distance between each term embed-

Class	#instances
Sustainable	1223
Unsustainable	1042
Total	**2265**

Table 2: Label distribution in the training set for sentence classification sub-task 02

ding and concept embedding is computed using the vector norm of the difference between the two embeddings. For each term, scores for all concepts are computed and the top k concepts are used as predicted concepts.

- Baseline 2 (B_2): A Word2Vec model trained on the given reports is used to generate term embeddings. Next, a Logistic Regression classifier is trained using these embeddings to do multi-class classification over the concepts. The final model consists of a Word2Vec model as the encoder and the trained Logistic Regression classifier as the decoder.

- Pre-trained DistilBERT (DistilBERT^P): This baseline is similar to Baseline 1 except that a pre-trained DistilBERT-base model is used as the encoder.

- Fine-tuned DistilBERT (DistilBERT^F): A pre-trained DistilBERT model was further fine-tuned on the sentences from the reports using the Masked Language Modelling task. The aim of this baseline is to see if training on the sentences in the given reports results in richer term and concept embeddings.

- Pre-trained Sentence-BERT (SentBERT^P): A pre-trained Sentence-BERT paraphrase detector (paraphrase-MiniLM-L3-v2) is used as the encoder to generate term and concept embeddings. The generated embeddings are then used to compute cosine distances between a term and all concepts. The top k ranked concepts are then selected as the predicted concepts.

- Pre-trained Sentence-BERT + Logistic Regression (SentBERT$^P_{LR}$): This baseline is similar to Baseline 2 except that a pre-trained Sentence-BERT paraphrase detector is used as the encoder to obtain term and concept embeddings.

Baseline	Accuracy	Mean Rank
Baseline 1	0.47	2.27
DistilBERT^P	0.34	2.72
DistilBERT^F	0.45	2.28
SentBERT^P	0.56	2.04
Baseline 2*	0.76	1.46
SentBERT$^{P*}_{LR}$	**0.79**	**1.41**

Table 3: Statistics showing the results of various baselines for sub task 01. First four scores are reported on the training set with no training. The last two models marked with * report the average scores with 10-fold cross validation on the training set.

For pre-trained DistilBERT and Sentence-BERT baselines, numerous variants were tested in the same setting for each of the baselines. However, we only report the best of the variants here due to space restrictions. We also tried using an approach similar to Wang et al. (2021) which encodes corrupted sentences into fixed-sized vectors and requires the decoder to reconstruct the original sentences from this sentence embedding, using RoBERTa as the encoder and decoder, on the sentences from the given reports to learn embeddings. Using this encoder to get embeddings, we train a Logistic Regression classifier, which gave similar performance to the baselines, and the model did not learn anything from the auto-encoder recontruction on the sentences from the reports to learn better embeddings.

Table 3 shows the results of the initial experiments and that SentBERT$^P_{LR}$ gave the best accuracy of 0.79 and a mean rank of 1.41.

5.2 Derived Dataset

In the SentBERT$^P_{LR}$ system, the weights of the Logistic Regression model are learnt during the training phase. There is no change in the weights of the Sentence-BERT model, thus, the training process has no impact on the generated embeddings. In order to enrich the generated embeddings, we propose training the encoder on a simple task of The following steps were followed for creating the derived dataset:

1. Obtain top 5 concept predictions for each term in the train set using the SentBERT^P model.

2. From the predictions create a dataset containing positive and negative samples.

3. A positive sample is the correct term-concept

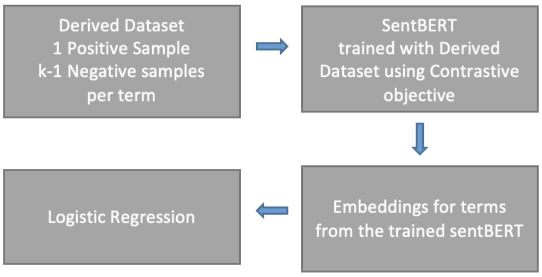

Figure 1: Proposed overall model for sub task 01

mapping.

4. A negative sample is a mapping between a term and an incorrectly predicted concept in the top k predictions.

5.3 System Description

The initial experiments using SentBERT^P show that although the embeddings generated by the model are richer, there is still room for improvement. The model, trained on paraphrase detection data, manages to capture the *hypernym* relation to some extent. If further fine-tuning of the model is carried out, it should ensure two things - correct neighbourhood relationship between term and concept embedding vectors in the current embedding space should be maintained, and missing neighbourhood relationships between correct term-concept vectors should be established. Previous work of Hadsell et al. (2006) proposed a contrastive loss function for this task. Contrastive loss given by equation 1. Here Y is the label of an instance, D_W is the distance between the concept and the term. The first section of the addition on the right side of the equation relates to the scenario when the model sees a positive example. The second section of the addition relates to the scenario when a negative example is seen. The constant m is the margin around the term within which a concept is considered a valid mapping. For all experiments, the value of m was set to 0.5.

$$L = (1-Y)\frac{1}{2}(D_W)^2+(Y)\frac{1}{2}\{max(0, m-D_W)\}^2 \quad (1)$$

The trained SentBERT model, SentBERT^F is then used with a Logistic Regression classifier as shown in figure 1. The system takes a term as input, generates term embedding using SentBERT^F as the encoder, and uses the embedding and a Logistic Regression classifier to predict the concept class.

Baseline	Accuracy	Mean Rank
SentBERT$^P_{LR}$	0.79 (Avg.)	1.41 (Avg.)
fold-0	0.8	1.46
fold-1	0.81	1.38
fold-2	0.70	1.61
fold-3	0.84	1.24
fold-4	0.75	1.55
fold-5	0.81	1.41
fold-6	0.80	1.38
fold-7	0.90	1.1
fold-8	0.78	1.5
fold-9	0.73	1.51
SentBERT$^F_{LR}$	**0.89** (Avg.)	**1.24** (Avg.)
fold-0	0.83	1.43
fold-1	0.90	1.2
fold-2	0.86	1.36
fold-3	0.90	1.21
fold-4	0.93	1.18
fold-5	0.92	1.15
fold-6	0.86	1.27
fold-7	0.93	1.09
fold-8	0.87	1.26
fold-9	0.87	1.28

Table 4: Statistics showing the impact of fine-tuning the SentBERT^P model on the derived dataset for sub task 01. The experiments were carried out with 10-fold cross validation.

5.4 Results and Analysis

Table 4 show the results of SentBERT$^F_{LR}$ on 10-fold training dataaset. Fine-tuning the SentBERT^P model results in a 10% increase in the average accuracy of the previous best model. This increase also results in a 0.17 reduction in the mean rank across 10-folds. The predictions obtained on the test set using a model trained on a random fold were submitted as part of the shared task. The predictions received an accuracy of 0.6 and a mean rank of 1.97. At this point, test labels have not been released and thus, error analysis cannot be carried out on the test set resulting in the usage of the validation set for a single fold for error analysis.

For error analysis, the fold with the lowest accuracy on the corresponding fold test set was used (fold-0). The size of the test set for fold-0 is 65 and of these 13 (20%) were classified incorrectly. Table 5 shows the distribution of the test set in terms of concepts and of these how many were incorrect.

Concept	Total Count	Incorrect Count
Energy efficiency and renewable energy	10	**4**
Board Make-Up	6	0
Carbon factor	5	**2**
Executive compensation	5	**2**
Product Responsibility	5	**1**
Sustainable Food & Agriculture	4	0
Shareholder rights	4	0
Employee engagement	4	0
Community	3	**1**
Emissions	3	0
Human Rights	2	**1**
Waste management	2	0
Biodiversity	2	0
Sustainable Transport	2	0
circular economy	2	0
Water & waste-water management	2	0
Injury frequency rate for subcontracted labour	2	**2**
Future of work	1	0
Employee development	1	0

Table 5: Concept distribution of the test set instances along with the corresponding counts for number of instances that were incorrectly classified in sub task 01.

Of the 17 concepts in the train set, 7 concepts had incorrectly classified instances. Figure 2 shows the confusion matrix for the incorrectly predicted classes. From the confusion matrix it can be seen that the model primarily has difficulty in understanding the difference between Emissions and the concepts *Energy efficiency and renewable energy* and *Carbon factor*.

6 Sentence Classification

In sub-task 2, we holdout 20 percent of the data (463 instances of 2265) as validation set to evaluate performance of our various approaches and fine-tune the hyperparameters. We use rest of the data

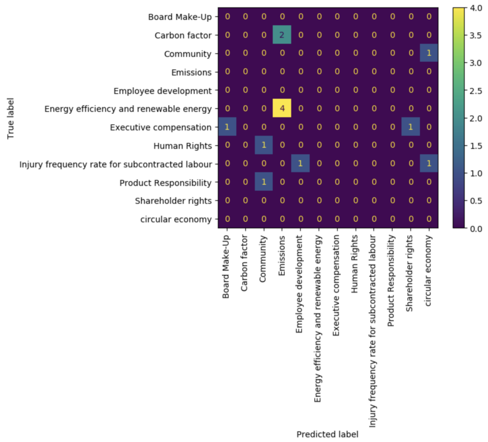

Figure 2: Confusion matrix for the incorrectly predicted classes.

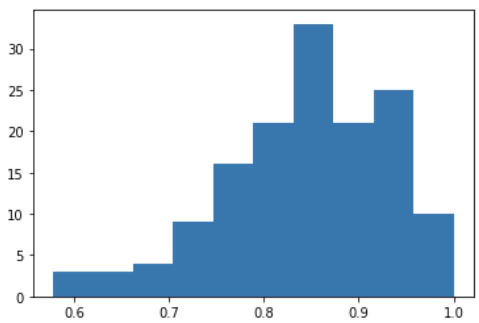

Figure 3: Histogram plot of Pair wise similarity for sentences in the train set with the test set in sub task 01.

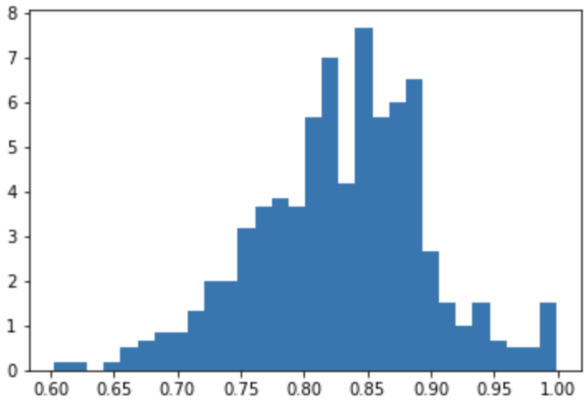

Figure 4: Histogram plot of Pair wise similarity for sentences in Val set with train set in sub task 02.

for training. We have built the following systems for sub task 02:

- Baseline 1 (B_1): We generate Term Frequency and Inverse Document frequency for the given data. Next, a Logistic Regression classifier is trained to perform binary classification.
- Baseline 2 (B_2): This baseline is similar to Baseline 1 except that a Naive Bayes model is used as the classifier.
- Leveraging Pretrained LMs: The world of NLP has extensively benefited from the development of large pretrained Language Models(LMs). Architectures such as ELMO (Peters et al., 2018), various extensions of BERT (Devlin et al., 2018; Liu et al., 2019b), XLNET (Yang et al., 2019), GPT (Brown et al., 2020), T5 (Raffel et al., 2019), etc have demonstrated dramatic improvements over conventional approaches. We were interested in leveraging such pretrained LMs in identifying if the given sentence is sustainable or unsustainable. To accomplish this we have built multiple systems where we finetune a pretrained LM using the data from sub task 02, as can be seen in table 6.

6.1 Discussion

As can be seen from the results in table 6, RoBERTa based model achieves the best performance among all the approaches we have tried. Using the Sentence Bert (Reimers and Gurevych, 2019) employed for sub task 01, we calculate the pairwise similarity between all the sentences of train set and

Model	Accuracy	Precision	Recall	F1
Baseline 01	85	85	86	85
Baseline 02	77.26	83.9	75.42	75.03
BERT	92.4	92	92	92
T5	93.3	93.5	93.3	93.3
RoBERTa	**96**	**96**	**96**	**96**

Table 6: Statistics showing the results on Val set for various models for Subtask 02.

held out validation set. The histogram plot of the similarity can be seen in figure 4. Here is an example pair of sentences from train and val sets that has high similarity score(0.91):

- *Val Sentence*: In 2020, as part of our **commitment** to carbon neutrality, we began focusing Scope 2 REC purchases on a country-by-country basis, depending on where the electricity is being used.
- *Train Sentence*: In 2020, as part of our **approach** to carbon neutrality, we began focusing Scope 2 REC purchases on a country-by-country basis, depending on where the electricity is **actually** being used.

It has to be noted that these sentences differ only in the words highlighted in bold and are almost identical to each other. Since the sentences seem very similar across the train and val sets, we were interested in seeing if the model was biased towards sentences it has already seen during training. To alleviate this and further validate our results from pretrained LMs, we performed 10 fold cross validation to prevent model over fitting to a section of training data. The results from cross validation can be found in table 7. We have submitted this system to the shared task and obtained joint third position on the leader board with accuracy of 92.6 percent.

6.2 Error Analysis

To understand the type of errors being made by our model, we have performed word level attribute analysis on the trained model. For this, we have used the open source package transformers-interpret[1]. Here are the types of errors being made by our model.

[1]https://github.com/cdpierse/transformers-interpret

All these activities help the supply chain to reduce its carbon footprint .

This year , the carbon offset ting initiative will be implemented by the Indonesian non - governmental organisation ID EP Foundation .

(a) Errors due to missed temporal modeling

We have an important role to play in achieving low - carbon operations in line with climate science .

As both a major user of energy and a producer of technologies that are essential to a lower - carbon economy , we have a responsibility to act .

(b) Errors due to bias on adjectives

Learn more about our efforts to reduce our scope 3 emissions in the Climate action chapter .

We are currently setting up processes to record non - reported Scope 3 data more precisely

(c) Errors due to Insufficient Information

Energy use within our offices accounts for 21 % of our carbon footprint .

(d) Errors due to logical inconsistency

Then all packaging waste is recycled or recovered , and contributes to the improvement of the ratio of waste recovery .

(e) Other Errors

Figure 5: Error Analysis - Categorization of errors made by our model for sub task 02.

- *Errors due to missed Temporal Modeling*: These are the errors due to the model being unaware of the temporal context of a sentence. Examples of this type of errors are given in (a) of figure 5.

- *Errors due to bias on Adjectives*: We have noticed that attention in our model is biased towards adjective words which might be misleading the prediction when the context is ambiguous. Examples of this type of errors are given in (b) of figure 5.

- *Errors due to insufficient information*: There are sentences that lack the information required to make a prediction even for humans. We depict examples of this error type in (c) of figure 5.

- *Errors due to logical inconsistency*: There are a few errors where the model misses the logical consistency. For instance, in the example shown in (d) of figure 5, the model considers 21 as a positive attribute towards making the decision.

- Other Errors: Example of this type of errors are mentioned in (e) of figure 5.

6.3 Observations

The sentences in test and train sets have high degree of similarity. There are instances where the sentences are nearly identical as mentioned in the discussion sub section. In addition, there are also sentences which are paraphrases of each other. Here is an example pair of sentences from train and test sets:

- *Train Sentence:* Our operational carbon footprint (occupied offices and business travel) will be net zero from 2030.

- *Test Sentence:* From 2030, our operational footprint (occupied offices and business travel) will operate with net zero carbon emissions.

Given the high levels of similarity, we hypothesize that architectures that can model paraphrasing can perform well on this sub task. It might be interesting to employ models that can generate paraphrases of original sentences to augment the

Fold	Accuracy	Precision	Recall	F1
Fold 01	95	95	96	95
Fold 02	94	94	93	93
Fold 03	92.4	92	92	92
Fold 04	93.3	93.5	93.3	93.3
Fold 05	96	96	96	96
Fold 06	95	95	96	95
Fold 07	95	95	95	94
Fold 08	91.4	91	91	91
Fold 09	93.3	93.5	93.3	93.3
Fold 10	96	96	96	96

Table 7: Results of 10 fold Cross Validation using Roberta Model on Subtask 02

Task	Accuracy	Mean Rank
Sub Task 01	60.08	1.97
Sub Task 02	92.68	-

Table 8: Test Results of our submissions to the shared task.

training data and achieve competitive performance even in low resource scenarios.

7 Test Submission

As part of the shared task, we have made submissions to both the subtasks. Our team name is Jetsons and we have presented the results of our systems from both sub tasks in the table 8. We are nearly 24 percentage points off from the best system in sub task 01. We are in joint third position in sub task 02.

8 Conclusion

In this paper, we presented our submission to the sub tasks of FinSim4-ESG. We first present a system that addresses the taxonomy enrichment problem for “Environment, Social and Governance" issues in the financial domain. We first created a derived dataset for taxonomy enrichment by using a sentence-BERT-based paraphrase detector to create positive and negative term-concept pairs. We employ a Logistic Regression classifier as the decoder, resulting in test Accuracy: 0.6 and Avg. Rank: 1.97. We then present our approach to the sub task of sentence classification. Our best performing model, a finetuned version of RoBERTa model achieves 96 percent on validation set and 92.3 on test set.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE.

David Jurgens and Mohammad Taher Pilehvar. 2016. Semeval-2016 task 14: Semantic taxonomy enrichment. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 1092–1102.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019a. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Irina Nikishina, Natalia Loukachevitch, Varvara Logacheva, and Alexander Panchenko. 2021. Evaluation of taxonomy enrichment on diachronic wordnet versions. In *Proceedings of the 11th Global Wordnet Conference*, pages 126–136.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations." arxiv preprint. *arXiv preprint arXiv:1802.05365*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021. Tsdae: Using transformer-based sequential denoising auto-encoder for unsupervised sentence embedding learning.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Qingkai Zeng, Jinfeng Lin, Wenhao Yu, Jane Cleland-Huang, and Meng Jiang. 2021. Enhancing taxonomy completion with concept generation via fusing relational representations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2104–2113.