Don't Prompt, Search!
Mining-based Zero-Shot Learning with Language Models

Mozes van de Kar[1] **Mengzhou Xia**[2] **Danqi Chen**[2] **Mikel Artetxe**[3]
[1]University of Amsterdam [2]Princeton University [3]Meta AI
mozesvandekar@gmail.com {mengzhou,danqic}@cs.princeton.edu
artetxe@meta.com

Abstract

Masked language models like BERT can perform text classification in a zero-shot fashion by reformulating downstream tasks as text infilling. However, this approach is highly sensitive to the template used to prompt the model, yet practitioners are blind when designing them in strict zero-shot settings. In this paper, we propose an alternative mining-based approach for zero-shot learning. Instead of prompting language models, we use regular expressions to mine labeled examples[1] from unlabeled corpora, which can optionally be filtered through prompting, and used to finetune a pretrained model. Our method is more flexible and interpretable than prompting, and outperforms it on a wide range of tasks when using comparable templates. Our results suggest that the success of prompting can partly be explained by the model being exposed to similar examples during pretraining, which can be directly retrieved through regular expressions.

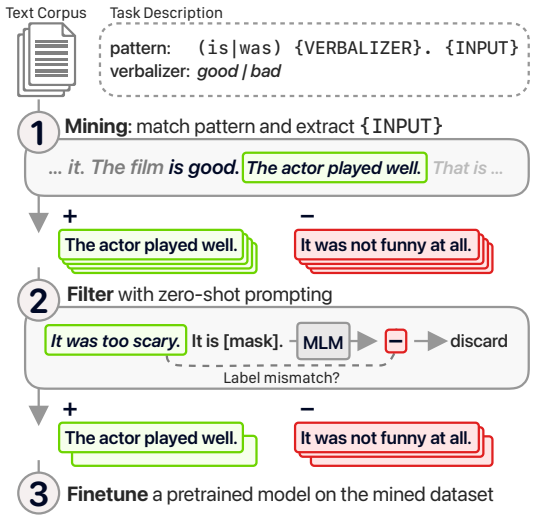

Figure 1: **Proposed method.** 1) We mine labeled examples from a text corpus with regex-based patterns. 2) Optionally, we filter examples for which zero-shot prompting predicts a different label. 3) We finetune a pretrained language model with a classification head.

1 Introduction

Recent work has obtained strong zero-shot results by prompting language models (Brown et al., 2020; Chowdhery et al., 2022). As formalized by Schick and Schütze (2021a), the core idea is to reformulate text classification as language modeling using a *pattern* and a *verbalizer*. Given the input space X, the output space C and the space of possible strings V^*, the pattern $t : X \rightarrow V^*$ maps each input into a string with a masked span, whereas the verbalizer $v : C \rightarrow V^*$ maps each label into a string. A language model can then be used for zero-shot classification by picking the most likely completion for the masked text $\arg\max_{c \in C} p(v(c) \mid t(x))$.[2] In few-shot settings, better results can be obtained by prepending a few labeled examples (Brown et al., 2020), or using them in some form of fine-tuning (Schick and Schütze, 2021a; Gao et al., 2021).

However, prompting is known to be sensitive to the choice of the pattern and the verbalizer, yet practitioners are blind when designing them in true zero-shot settings (Jiang et al., 2020; Perez et al., 2021). Connected to that, subtle phenomena like the surface form competition (Holtzman et al., 2021) have a large impact on performance. Recent work has tried to mitigate these issues through calibration (Zhao et al., 2021), prompt combination (Schick and Schütze, 2021a; Lester et al., 2021; Zhou et al., 2022) or automatic prompt generation (Shin et al., 2020; Gao et al., 2021). At the same time, there is still not a principled understanding of how language models become few-shot learners, with recent work analyzing the role of the pretraining data (Chan et al., 2022) or the input-output mapping of

[1]We use 'labeled examples' throughout the paper to denote the examples that match regex-based patterns of different labels. They are *weakly-supervised* and can be noisy.

[2]We focus on masked language models, and allow multi-token verbalizers through autoregressive decoding (see §3). Left-to-right language models also fit the framework by placing the mask at the end or scoring the full populated prompt.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 7508 - 7520
December 7-11, 2022

Task	Prompting pattern	Mining pattern
Sentiment	`{INPUT}. It was {VERBALIZER}.`	`(is|was) {VERBALIZER}*. {INPUT}`
Topic class.	`{INPUT}. It is about {VERBALIZER}.`	`{VERBALIZER}*. {INPUT}`
NLI	`{INPUT:HYP} {VERBALIZER}, {INPUT:PREM}`	`{INPUT:HYP} {VERBALIZER}, {INPUT:PREM}`

Table 1: **Patterns.** `{VERBALIZER}` is replaced with the verbalizers in Table 2. For mining, `*.` captures everything up to a sentence boundary, and `{INPUT}`, `{INPUT:HYP}` and `{INPUT:PREM}` capture a single sentence.

Task	Lbl	Verbalizers
Sent.	Pos.	good, great, awesome, incredible
	Neg.	bad, awful, terrible, horrible
NLI	Ent.	Yes, Therefore, Thus, Accordingly, Hence, *For this reason*
	Con.	No, However, But, *On the contrary*, *In contrast*
	Neu.	Maybe, Also, Furthermore, Secondly, Additionally, Moreover, *In addition*

Table 2: **Verbalizers for sentiment classification and NLI.** See Table 9 for verbalizers used in topic classification. When using a single verbalizer, we choose the one underlined. Multi-token verbalizers are in italic. Lbl: label, Ent./Con./Neu: entailment, contradiction, neutral.

in-context demonstrations (Min et al., 2022).

In this paper, we propose an alternative approach to zero-shot learning that is more flexible and interpretable than prompting, while obtaining stronger results in our experiments. Similar to prompting, our method requires a pretrained language model, pattern, and verbalizer, in addition to an unlabeled corpus (e.g., the one used for pretraining). As illustrated in Figure 1, our approach works by using the pattern and verbalizer to mine labeled examples from the corpus through regular expressions, and leveraging them as supervision to finetune the pretrained language model. This allows to naturally combine multiple patterns and verbalizers for each task, while providing a signal to interactively design them by looking at the mined examples. In addition, we show that better results are obtained by filtering the mined examples through prompting.

Experiments in sentiment analysis, topic classification and natural language inference (NLI) confirm the effectiveness of our approach, which outperforms prompting by a large margin when using the exact same verbalizers and comparable patterns. Our results offer a new perspective on how language models can perform downstream tasks in a zero-shot fashion, showing that similar examples often exist in the pretraining corpus, which can be directly retrieved through simple extraction patterns.

2 Proposed Method

As shown in Figure 1, our method has three steps:

Mine. We first use the pattern and a set of verbalizers to extract labeled examples from the corpus. To that end, we define patterns that are filled with verbalizers and expanded into regular expressions. For instance, the pattern and verbalizer in Figure 1 would extract every sentence following *"is good."* or *"was good."* as an example of the positive class, and every sentence following *"is bad."* or *"was bad."* as an example of the negative class. In practice, the patterns that we define are comparable to the ones used for prompting, and the verbalizers are exactly the same (see Tables 1 and 2). Appendix A gives more details on how we expand patterns into regular expressions. While prior work in prompting typically uses a single verbalizer per class, our approach allows to naturally combine examples mined through multiple verbalizers in a single dataset. So as to mitigate class imbalance and keep the mined dataset to a reasonable size, we mine a maximum of 40k examples per class after balancing across the different verbalizers.

Filter. As an optional second step, we explore automatically removing noisy examples from the mined data. To that end, we classify the mined examples using zero-shot prompting, and remove examples for which the predicted and the mined label do not match. This filtering step is reliant on the performance of prompting, and we only remove 10% of the mismatching examples for which zero-shot prompting is the most confident.

Finetune. Finally, we use the mined dataset to finetune a pretrained language model in the standard supervised fashion (Devlin et al., 2019), learning a new classification head.

3 Experimental Settings

Tasks. We evaluate on three types of tasks: *binary sentiment analysis* on Amazon (Zhang et al.,

		Sentiment analysis					Topic class.			NLI				avg
		amz	imd	mr	sst	ylp	agn	dbp	yah	mnl	qnl	rte	snl	
Full-shot	Fine-tuning	97.1	95.7	88.8	94.4	95.0	95.1	99.3	76.8	78.5	92.6	67.6	90.5	89.3
Zero-shot	Prompting	81.5	78.4	71.1	77.4	81.9	34.0	36.4	28.2	47.1	50.8	52.3	39.6	56.6
	w/ multi verb.	83.5	81.8	78.3	81.9	83.1	54.6	51.1	34.1	46.5	**58.2**	61.4	44.1	63.2
	Proposed method	**92.0**	**86.7**	**80.5**	**85.6**	**92.0**	**79.2**	**80.4**	**56.1**	**50.4**	53.2	**62.6**	**46.0**	**72.0**

Table 3: **Main results (accuracy).** All systems are based on RoBERTa-base, and all zero-shot systems use comparable patterns (see Table 1). We report average accuracy across 3 runs for all systems except prompting. w/ multi verb.: prompting with different sets of verbalizers (Table 9) and averaging the probabilities.

2015), IMDb (Maas et al., 2011), MR (Pang and Lee, 2005), SST-2 (Socher et al., 2013) and Yelp (Zhang et al., 2015), *topic classification* on AG News (Zhang et al., 2015), DBPedia (Zhang et al., 2015) and Yahoo Topics[3] (Zhang et al., 2015), and *NLI* on MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), RTE (Dagan et al., 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009) and SNLI (Bowman et al., 2015). We report accuracy on the test set when available, falling back to the validation set for SST-2, MNLI, RTE and QNLI. For all systems involving fine-tuning, we report the average across 3 runs with different random seeds. We ran all development experiments on SST-2 and AG News without any exhaustive hyperparameter exploration, and evaluate the rest of the tasks blindly.

Approaches. We compare the following methods in our experiments, using RoBERTa-base (Liu et al., 2019) as the pretrained model in all cases:

- **Full-shot fine-tuning**: We finetune RoBERTa on the original training set adding a new classification head. We train for 3 epochs with a batch size of 32. All the other hyperparameters follow Liu et al. (2019). Refer to Appendix B for more details.

- **Zero-shot prompting**: Standard prompting, described in §1. Multi-token verbalizer probabilities are calculated autoregressively, picking the most likely token at each step (Schick and Schütze, 2021c). We report results using both a single verbalizer per class, as it is common in prior work, as well as multiple verbalizers per class, which is more comparable to our approach. For the latter, we combine the probabilities of each verbalizer by averaging.[4]

- **Zero-shot mining**: Our proposed method, described in §2. For the mining step, we use the first 100 shards from the C4 corpus (Raffel et al., 2020), which cover 9.8% of the data. For the filtering step, we use single-verbalizer prompting to filter 10% of the mislabeled examples. For the fine-tuning step, we use the same settings as in the full-shot setup, except that we train for 5,000 steps with a dropout probability of 0.4.[5] To mitigate class imbalance, we form batches by first sampling the class for each instance from the uniform distribution, and then picking a random example from the mined data belonging to that class.

[3]The Yahoo Answers dataset was downloaded by, and access was limited to, the University of Amsterdam, where all experiments were carried out.

	Prompting	Mining
good / *bad*	**78.1**	72.0
great / *awful*	**82.3**	82.1
awesome / *terrible*	82.3	**83.9**
incredible / *horrible*	83.1	**87.3**
combined	81.7	**85.4**

Table 4: **Average sentiment accuracy using different verbalizers.** We report mining results without filtering. More detailed results are provided in Table 12.

Patterns and verbalizers. We use comparable patterns for prompting and mining with the exact same verbalizers, which we report in Table 1 and 2. These were designed without any experiment, simulating a zero-shot setting. We design our patterns

[4]We also tried summing or taking the maximum, which obtained similar results as shown in Appendix C.

[5]During development, we found that high dropout and early stopping help mitigating model overfitting caused by the misalignment between the mined and the true distribution. However, evaluation on all tasks shows mixed results. We stick to the original setup with high dropout to be faithful to the rigorous zero-shot scenario, and report additional results with standard dropout in Appendix C.

Pattern	Verbalizer	avg
Prompting		
`{VERBALIZER} stars: {INPUT}`	*5 / 1*	51.0
`{INPUT} I {VERBALIZER} it.`	*love / hate*	73.1
`{INPUT} It is {VERBALIZER}.`	*good / bad*	**78.1**
Mining		
`{VERBALIZER} star*. {INPUT}`	*5 / 1*	**72.1**
`I {VERBALIZER}*. {INPUT}`	*love / hate*	**85.5**
`(is|was) {VERBALIZER}*. {INPUT}`	*good / bad*	72.0

Table 5: **Average sentiment accuracy using different patterns and verbalizers.** We report mining results without filtering (more details are provided in Table 13).

Data	Filter	Sent.	Topic	NLI
Supervised	-	94.2	90.4	82.3
Mined	none	85.4	71.5	52.0
	prompting	87.4	71.9	53.0
	supervised†	91.4	85.5	63.8

Table 6: **Filtering results (average accuracy).** †: uses mined data for training and another supervised classifier as the filter. This is not a zero-shot setting and serves as an upper limit for the results using a perfect filter. More detailed results are provided in Table 11.

to capture sentences following a verbalizer, rather than sentences containing the verbalizer, as the resulting dataset would otherwise be trivial (solvable by detecting the presence of certain words).

4 Results and Analysis

We next discuss our main findings and report additional results in Appendix C.

Main results. We report our main results in Table 3. Our method outperforms prompting by 8.8 points on average, and the improvements are consistent across all tasks.

Effect of patterns and verbalizers. Table 4 reports sentiment results using different verbalizers. Consistent with prior work, we find that both prompting and mining are highly sensitive to the choice of the verbalizer, yet combining them all roughly matches the results of the best performing one. As shown in Table 5, using different patterns has an even larger impact. Interestingly, patterns and verbalizers that do well with one approach do not necessarily do well with the other.

Effect of filtering. Table 6 reports additional results using the full-shot systems for filtering, or not using any filtering at all. We find that prompting-based filtering brings modest but consistent improvements across all types of tasks. We compare this to filtering out all examples with mismatching labels with the full-shot model, which results in much larger gains and approaches the performance of the fully supervised system for sentiment and topic classification tasks. This can be seen as an upper-limit of what could be reached with perfect filtering, which leaves ample room to improve our approach focusing on the filtering step alone.

#	Lbl	Mined example
1	Pos.	Do you have an idea of how broad your vocal range was?
2	Pos.	Once home, we began priming.
3	Neg.	People in Wall Street and other financial services firms should have paid more attention to the data.
4	Neg.	So I bought this unit, which said it had the same technical features as the other brand, such as number of channels etc, and this one performed amazing!!

Table 7: **Mined examples** for sentiment analysis. See more examples in Table 17 and mined NLI examples in Table 18.

Qualitative analysis. We manually assessed 20 mined examples for sentiment analysis and report some representative instances in Table 7. We find that the mined data covers many domains like finance and technology. Most examples are correct (#1, #3), but there are also instances with wrong labels (#4). In addition, we find that 40% of analyzed examples show weak or neutral sentiment (#2). The impact of such irrelevant examples is unclear and worth of future study.

5 Related work

Recent work in zero-shot learning has explored a similar *generate-filter-finetune* approach, but using large language models instead of mining to generate training data (Schick and Schütze, 2021b; Liu et al., 2022; Meng et al., 2022; Ye et al., 2022). Mining-based approaches have a long tradition in information extraction (Riloff, 1996; Riloff and Jones, 1999). However, to the best of our knowledge, we are the first to apply them for zero-shot learning as an alternative to prompting. Instead of mining examples for the target task, Bansal et al. (2020) define task-agnostic pretraining objectives on unlabeled corpora. Closer to our work, Meng et al. (2020) mask label-indicative words in an

unlabeled corpus, and train a model to predict their corresponding label. Concurrent to our work, Han and Tsvetkov (2022) try locating a subset of the pretraining data that supports prompting in specific tasks. Finally, Razeghi et al. (2022) show a strong correlation between performance on specific instances and the frequency of terms from those instances in the pretraining data.

6 Conclusions

In this work, we have shown that mining-based zero-shot learning outperforms prompting. Moreover, our approach shows headroom for further improvement by exploring filtering techniques. The flexibility of our approach enables additional directions like domain filtering, bootstrapping, and interactive pattern/verbalizer design, where practitioners would inspect a few mined examples and refine their patterns until they are satisfied. In addition, our methods can serve as a partial explanation for why prompting works, showing that task-relevant examples are often present in the pretraining corpus in an explicit form, to the extent that they can be directly mined through simple regular expressions. Nevertheless, we believe that there can be other factors involved, as evidenced by the best patterns and verbalizers being different for mining and prompting, and we believe that delving deeper into the relation between pretraining data and prompting performance is an interesting future direction.

Limitations

Developing zero-shot methods in a rigorous manner is challenging: the strict zero-shot scenario does not allow using annotated data except for the final evaluation, yet it is difficult to make development decisions without any signal. We decided to use AG News and SST-2 during development without any exhaustive hyperparameter exploration, and evaluate blindly in the rest of the tasks. At the same time, we designed all patterns and verbalizers without any experiment, based solely on our own intuition. We believe that the comparison between prompting and mining is fair as we used comparable patterns with the exact same verbalizers and pretrained model. However, it is possible that our patterns, verbalizers and/or hyperparameters are suboptimal, and better results could be obtained with either prompting or mining using other configurations.

An important limitation of our approach is that it can be difficult to design extraction patterns for certain tasks like multiple choice questions. However, prompting is known to suffer from a similar limitation, with certain tasks like WiC being difficult to formulate as language modeling and obtaining random chance performance (Brown et al., 2020).

Different from prompting, our approach requires an intermediate step after pretraining to mine data and finetune the model, which takes 2-7 hours using a single Nvidia Titan RTX GPU and 4 Intel Xeon CPUs. However, inference cost is similar or even faster than prompting, as our approach does not incur on any overhead for multi-token and multi-verbalizer setups.

Acknowledgements

We thank Ves Stoyanov, Jingfei Du, Timo Schick and Sewon Min for their feedback. Mozes van de Kar received a travel grant from ELLIS and Qualcomm to attend the conference.

References

Trapit Bansal, Rishikesh Jha, Tsendsuren Munkhdalai, and Andrew McCallum. 2020. Self-supervised meta-learning for few-shot natural language classification tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 522–534, Online. Association for Computational Linguistics.

Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge. In *Proceedings of the second PASCAL challenges workshop on recognising textual entailment*, volume 6. Venice.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2009. The fifth pascal recognizing textual entailment challenge. In *TAC*.

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, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec

Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Stephanie C. Y. Chan, Adam Santoro, Andrew K. Lampinen, Jane X. Wang, Aaditya Singh, Pierre H. Richemond, Jay McClelland, and Felix Hill. 2022. Data distributional properties drive emergent in-context learning in transformers.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

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

Xiaochuang Han and Yulia Tsvetkov. 2022. Orca: Interpreting prompted language models via locating supporting data evidence in the ocean of pretraining data. *arXiv preprint arXiv:2205.12600*.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. 2021. Surface form competition: Why the highest probability answer isn't always right. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7038–7051, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022. Wanli: Worker and ai collaboration for natural language inference dataset creation.

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

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. 2022. Generating training data with language models: Towards zero-shot language understanding.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. 2020. Text classification using label names only: A language model self-training approach. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9006–9017, Online. Association for Computational Linguistics.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work?

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. In *Advances in Neural Information Processing Systems*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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

Yasaman Razeghi, Robert L. Logan, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot reasoning.

Ellen Riloff. 1996. Automatically generating extraction patterns from untagged text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2*, AAAI'96, page 1044–1049. AAAI Press.

Ellen Riloff and Rosie Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, AAAI '99/IAAI '99, page 474–479, USA. American Association for Artificial Intelligence.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021b. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6943–6951, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021c. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

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.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. Zerogen: Efficient zero-shot learning via dataset generation.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Tony Z. Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models.

Chunting Zhou, Junxian He, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2022. Prompt consistency for zero-shot task generalization.

A Pattern expansion for mining

For each class, examples are mined by filling in the pattern with the verbalizer and extracting sentences that match the filled-in pattern. The process of expanding the patterns into regular expressions is as follows. First, we replace `{VERBALIZER}` with a capturing group containing all verbalizers separated by the alternation operator `|`. For example, the verbalizer *good, great, awesome* is expanded into `(good|great|awesome)`. Finally, we replace the keywords described in Table 8 with the corresponding regular expressions. The result is a regular expression containing capturing groups for extracting sentences in a case-insensitive fashion.

Note that we use a simplistic sentence definition in order to keep the regex manageable. Since we assume that a period always ends a sentence, this mistakenly interprets abbreviations as multiple sentences (e.g., *"U.S.A."* contains 3 sentences). To address this, we filter out mined sentences shorter than 4 characters.

B Additional experimental details

Patterns and verbalizers. For each category of tasks we use the same mining pattern, as shown in Table 1. The complete list of verbalizers for each task is given in Table 9. Tasks with the same classes share the same verbalizers. This means that all sentiment and NLI tasks have the same verbalizers. Each topic classification task, however, has a unique set of verbalizers. Note that while SNLI and MNLI (3-way NLI) have the same verbalizers as RTE and QNLI (2-way NLI), the mined datasets do differ since 2-way NLI does not include a neutral class.

Hyperparameters. Table 10 shows the hyperparameters used for finetuning the RoBERTa-base model. All the other hyperparameters and classification head architecture follow Liu et al. (2019). We have two fine-tuning configurations, one for fine-tuning in the full-shot setting and one for zero-shot fine-tuning on the mined dataset. These configurations differ only in the maximum number of steps, dropout rate and batch sampler.

Datasets. We use Huggingface (Lhoest et al., 2021) for loading all evaluation datasets without any additional processing, except for MR which is detokenized using Moses scripts. We evaluate on the test set, falling back to the validation set for SST-2, MNLI, RTE and QNLI.

Keyword	Regex
`{VERBALIZER}`	Replaced with the verbalizer
`*`	regex: `[^.!?]*?` Greedily matches non-sentence-ending characters
`{INPUT}`	regex: `([^.!?]+[.!?]+)` Matches a single sentence, extracted with the key "INPUT"

Table 8: **Keywords that compile into regular expressions.** These keywords are used in the mining patterns and verbalizers.

C Additional results

Complete results for full-shot, prompting and mining are combined in Table 11. Results showing the effect of pattern and verbalizer choice on binary sentiment classification are presented in Table 12 and Table 13, respectively.

As explained in the main text, development experiments were only conducted on AGNews and SST-2. On these tasks, we found that high regularization partially mitigates overfitting caused by the misalignment between the mined dataset and real dataset. However, this high regularization shows mixed results for non-development tasks. For full transparency, we compare these performance differences in Table 14, but, in the main text, we stick to the original setup with high dropout to be faithful to the rigorous zero-shot scenario.

For multi-verbalizer prompting, we combine the probabilities of each verbalizer with an aggregation function. Results for using the average, the max and the sum are shown in Table 15.

In Table 16 we show the agreement between the mined labels and the labels according to the filtering method, which in our experiments is either a full-shot finetuned model or single-verbalizer prompting.

Table 17 and Table 18 show a random sample of examples from the mined training dataset for respectively binary sentiment analysis and NLI. In the main text, Table 7 shows a representative selection of examples for sentiment analysis. These examples were manually picked from the random sample in Table 17.

Task	Class	Verbalizers
Sentiment	Positive	good, great, awesome, incredible
	Negative	bad, awful, terrible, horrible
AGNews	World	world, foreign, global, Asia, Europe, China
	Sports	sports, football, basketball, tennis, soccer, baseball
	Business	business, stock, financial, profit, economy, finance
	Sci/Tech	technology, science, research, chemical, iPhone, smartphone
DBPedia	Company	company, business, manufacturer, *operates in*
	Educational institution	school, college, education, university
	Artist	artist, writer, song, composer
	Athlete	sports, runner, basketball, football
	Office holder	politics, president, Senate, politician
	Mean of transportation	bus, bike, car, train, ship, plane, aircraft
	Building	building, office, house, monument
	Natural place	river, forest hill, nature
	Village	town, village, *small population, small town*
	Animal	animal, species, horse, dog, pet, habitat
	Plant	plant, leaf, flower, herb
	Album	album,recording, *record company*
	Film	film, movie, actor, actress
	Written work	written, book, novel, poem
Yahoo	Society & Culture	culture, holiday, society
	Science & Mathematics	science, technology, math, research
	Health	health, body, exercise, *stress relieve*
	Education & Reference	school, college, education, university
	Computers & Internet	computer, internet, keyboard, software
	Sports	sports, football, basketball, game
	Business & Finance	business, stock, financial, profit
	Entertainment & Music	film, movie, actor, writer
	Family & Relationships	love, family, father, mother
	Politics & Government	politics, president, Senate, politician
NLI	Entailment	Yes, Therefore, Thus, Accordingly, Hence, *For this reason*
	Contradiction	No, However, But, *On the contrary, In contrast*
	Neutral	Maybe, Also, Furthermore, Secondly, Additionally, Moreover, *In addition*

Table 9: **Verbalizers.** When using a single verbalizer we choose the one underlined. In the multi-verbalizer setting we use all listed verbalizers. Sentiment includes Amazon, IMDB, MR, SST-2 and Yelp; NLI includes MNLI, QNLI, RTE and SNLI. Multi-token verbalizers are italic.

Parameter	full-shot	zero-shot
Model	RoBERTa-base (123M)	RoBERTa-base (123M)
Model selection	last	last
Batch size	32	32
Optimizer	adam	adam
Learning rate	1.00e-05	1.00e-05
LR schedule	6% warmup with linear decay	6% warmup with linear decay
Adam epsilon	1.00e-08	1.00e-08
Adam β_1	0.9	0.9
Adam β_1	0.999	0.999
Weight decay	0	0
Classifier dropout	0	0
Attention dropout	0.1	0.1
Hidden dropout	0.1	0.4
Max steps	-	5000
Max epochs	3	-
Batch sampler	-	inverse class frequency weighted sampling

Table 10: **Hyperparameters** for full-shot finetuning and zero-shot finetuning with the mined dataset.

		Full-shot	Prompting		Mining			
	maj	fine-tuning	single-verb	multi-verb	single-verb	multi-verb	+ filter full-shot	+ filter zero-shot
Sentiment Analysis								
Amazon	50.0	97.1±0.0	81.5	83.5	71.7±2.2	90.6±1.7	94.4±0.2	92.0±0.6
IMDB	50.0	95.7±0.0	78.4	81.8	78.2±0.5	83.0±4.1	91.6±0.4	86.7±1.3
MR	50.0	88.8±0.1	71.1	78.3	70.5±1.0	77.7±2.4	86.3±0.7	80.5±1.0
SST-2	50.9	94.4±0.2	77.4	77.4	77.3±1.4	83.8±2.4	89.5±0.7	85.6±1.1
Yelp	50.0	95.0±0.1	81.9	83.1	62.1±2.3	92.0±2.2	95.2±0.6	92.0±1.5
avg	50.2	94.2	78.1	81.7	72.0	85.4	91.4	87.4
Topic Classification								
AGNews	25.0	95.1±0.1	34.0	54.6	71.0±0.7	78.4±0.6	89.5±0.2	79.2±0.6
DBPedia	7.1	99.3±0.0	36.4	51.1	63.7±1.0	79.8±0.0	97.1±0.3	80.4±0.4
Yahoo	10.0	76.8±0.1	28.2	34.1	51.7±0.5	56.3±2.5	69.8±0.1	56.1±2.1
avg	14.0	90.4	32.9	46.6	62.1	71.5	85.5	71.9
NLI								
MNLI	35.3	78.50.0	47.1	46.5	48.2±0.5	49.2±0.5	65.5±0.3	50.4±0.4
QNLI	50.5	92.6±0.1	50.8	58.2	51.6±0.3	52.9±1.2	70.2±1.1	53.2±0.6
RTE	52.7	67.6±1.4	52.3	61.4	55.0±0.2	61.1±2.1	57.8±1.3	62.6±0.9
SNLI	34.3	90.5±0.1	39.6	44.1	38.0±0.7	44.6±0.9	61.9±2.8	46.0±1.1
avg	41.6	82.3	47.5	52.5	48.2	52.0	63.8	53.0
macro avg	38.6	89.3	56.6	63.2	61.6	70.8	80.7	72.1

Table 11: **Complete results for each task and system.** When applicable, results show fine-tuning average performance and standard deviation over 3 seeds. Full-shot shows the fine-tuning results with the hyperparameters described in Table 10. Prompting shows the single and multi-verbalizer baseline results. Mining results show single and multi-verbalizer performance without filtering, in addition to multi-verbalizer performance with full-shot and zero-shot filtering. Maj: majority baseline

	Amazon	IMDB	MR	SST-2	Yelp	avg
Prompting						
good / bad	81.5	78.4	71.1	77.3	81.9	78.1
great / awful	82.9	82.7	80.8	82.6	82.6	82.3
awesome / terrible	84.5	82.0	78.3	82.8	84.0	82.3
incredible / horrible	86.6	83.5	78.0	80.0	87.2	83.1
combined	83.5	81.8	78.3	81.9	83.1	81.7
Mining						
good / bad	71.7±2.2	78.2±0.5	70.5±1.0	77.3±1.4	62.1±2.3	72.0
great / awful	88.4±1.7	75.7±4.0	73.0±2.5	79.8±2.1	93.3±0.5	82.1
awesome / terrible	91.4±0.6	81.3±1.4	73.7±1.6	78.8±1.3	94.3±0.7	83.9
incredible / horrible	90.5±0.2	88.1±0.2	80.5±1.0	83.5±0.6	94.0±0.4	87.3
combined	90.6±1.7	83.0±4.1	77.7±2.4	83.8±2.4	92.0±2.2	85.4

Table 12: **Verbalizer comparison for sentiment tasks.** For mining, we report average performance and standard deviation over 3 seeds without filtering.

Pattern	Verbalizer	Amazon	IMDB	MR	SST-2	Yelp	avg
Prompting							
`{VERBALIZER} stars: {INPUT}`	*5 / 1*	50.4	50.0	50.0	50.9	53.7	51.0
`{INPUT} I {VERBALIZER} it.`	*love / hate*	77.6	73.3	64.6	69.8	80.3	73.1
`{INPUT} It is {VERBALIZER}.`	*good / bad*	81.5	78.4	71.1	77.4	81.9	78.1
Mining							
`{VERBALIZER} star*. {INPUT}`	*5 / 1*	68.7±0.4	75.4±2.1	70.8±1.7	78.3±2.4	67.5±5.3	72.1
`i {VERBALIZER}*. {INPUT}`	*love / hate*	88.9±0.3	84.0±1.2	79.2±0.8	84.0±0.8	91.1±0.7	85.5
`(is|was) {VERBALIZER}*. {INPUT}`	*good / bad*	71.7±2.2	78.2±0.5	70.5±1.0	77.3±1.4	62.1±2.3	72.0

Table 13: **Template comparison.** Performance for three different templates on sentiment tasks comparing prompting and mining without filtering. Additionally, we show standard deviations over three seeds for the mining approach. The verbalizer column shows the verbalizer for the positive and the negative class, respectively.

	Default dropout		High dropout	
	full-shot	zero-shot	full-shot	zero-shot
Sentiment Analysis				
Amazon	95.8±0.0	91.0±0.4	94.4±0.2	92.0±0.6
IMDB	94.4±0.2	80.1±4.0	91.6±0.4	86.7±1.3
MR	88.0±0.2	76.5±1.1	86.3±0.7	80.5±1.0
SST-2	92.0±0.5	80.2±1.6	89.5±0.7	85.6±1.1
Yelp	96.6±0.1	94.4±0.6	95.2±0.6	92.0±1.5
avg	93.3	84.4	91.4	87.4
Topic Classification				
AGNews	90.8±0.2	77.2±1.0	89.5±0.2	79.2±0.6
DBPedia	98.8±0.0	83.9±0.4	97.1±0.3	80.4±0.3
Yahoo	72.9±0.1	56.6±2.4	69.8±0.1	56.1±2.1
avg	87.5	72.5	85.5	71.9
NLI				
MNLI	77.7±1.0	52.9±0.8	65.5±0.3	50.4±0.4
QNLI	77.0±1.6	57.8±0.9	70.2±1.1	53.2±0.6
RTE	72.3±1.1	61.4±2.9	57.8±1.3	62.6±0.9
SNLI	80.2±0.2	49.4±1.0	61.9±2.8	46.0±1.1
avg	76.8	55.4±0.4	63.8	53.0
macro avg	86.4	71.8	80.7	72.1

Table 14: **Performance with high dropout and default dropout.** These results use our proposed mining method + filtering and compares 2 settings of the hidden layer dropout: the default setting of 0.1 and the high regularization setting of 0.4, the value that was found most effective during development experiments on AGNews and SST-2.

	Averaging	Max	Sum
Sentiment Analysis			
Amazon	83.5	82.7	83.5
IMDB	81.8	81.2	82.1
MR	78.3	77.4	78.3
SST-2	81.9	81.3	81.9
Yelp	83.1	82.3	83.1
Topic Classification			
AGNews	54.6	53.4	54.6
DBPedia	51.1	49.1	51.4
Yahoo	34.1	34.1	34.1
NLI			
MNLI	46.5	46.4	47.0
QNLI	58.2	58.4	57.4
RTE	61.4	61.7	60.6
SNLI	44.1	42.8	43.7

Table 15: Results for different **probability aggregation functions** for multi-verbalizer prompting.

	Full-shot	Prompting
Sentiment Analysis		
Amazon	68.7	63.6
IMDB	64.5	63.6
MR	66.1	63.6
SST-2	66.7	63.6
Yelp	69.1	63.6
Topic Classification		
AGNews	52.3	31.8
DBPedia	25.7	13.4
Yahoo	36.4	22.2
NLI		
MNLI	39.2	42.0
QNLI	52.7	62.2
RTE	50.3	62.2
SNLI	40.8	42.0

Table 16: **Label agreement.** Percentage of examples for which the mined label is equal to the label predicted by a full-shot model or by single-verbalizer prompting.

#	Mined Label	Mined Example
1	Negative	So I bought this unit, which said it had the same technical features as the other brand, such as number of channels etc, and this one performed amazing!!
2	Positive	The founders of Clickfunnels have focused not only on creating a great internet site for you yet also offering you enough expertise and details to act as an informed company person / entrepreneur.
3	Positive	Once home, we began priming.
4	Negative	While you can ' t view the content on the second page without either logging in or signing up since there ' s no ' x ' button, there ' s a trick involving 3D Touch that can help.
5	Positive	That i savored them a long way a great deal more compared to My spouse and i believed i would certainly.
6	Positive	Do you have an idea of how broad your vocal range was?
7	Positive	Also recently the lovely Megan Washington modelled for our latest transeasonal collection we just shot last week.
8	Positive	What are 3 things that make you happy?
9	Positive	Be devoted to one another in love.
10	Negative	An unexpected situation arose with my father requiring help for three to four months.
11	Positive	Simply AWESOME!
12	Negative	I don ' t disagree with that, however the voices have never been as loud or as many as now about this topic of "anti TPS ", that is progress, that is a movement, that is what we need to inspire EA to finally listen and do something about it.
13	Positive	Adverse drug reactions are based on evaluation of data from pre-marketing phase 2-3 studies and updated based on pooled data from 18 placebo-controlled pre- and post-marketing studies, including approximately 5,000 patients treated with varenicline.
14	Negative	I have the books from last year and have spoke to the college to get this years as they may be changing to another type.
15	Negative	I was down-to-my-core terrified.
16	Negative	He didn ' t win, and our support of him became rather limited when we determined he was not winning.
17	Positive	It was in a four-star hotel in the Boca Raton Resort Bungalows.
18	Positive	I wish my preschool was this nice.
19	Negative	People in Wall Street and other financial services firms should have paid more attention to the data.
20	Positive	I guess I am quite transparent.

Table 17: Random sample of mined examples for sentiment analysis.

#	Mined Label	Mined Premise	Mined Hypothesis
1	Neutral	When seniors and their caregivers develop and enjoy deep friendships, it can bolster feelings of well-being, happiness and security.	it can help to make activities like exercising and healthy eating more enjoyable.
2	Contradiction	Customer service skills training for staff should include basic customer service skills of active listening, how to handle difficult situations, telephone etiquette, and interpersonal communication tips.	most of all, it should focus on how to build a relationship with a student.
3	Entailment	The easy way to remember the arrhythmias most commonly associated with SSS — is to think of what one might expect if the SA node became "sick ".	there is sinus bradycardia and arrhythmia — sinus pauses (which may be longlasting, ultimately leading to sinus arrest) — and SA nodal block.
4	Contradiction	Primarily due to President Obama ' s historic announcement, more Americans are visiting Cuba, making a bad situation worse.	for those of you with the ability to book your Cuban Rent A Car in advance, all the above official websites still offer availability at the writing of this article.
5	Neutral	In 1989, Tufts University School of Medicine created the Minority High School Tutorial PLUS Program to provide local minority / disadvantaged students with access to medical student tutors.	in 1989, Tufts University School of Medicine received a grant from the National Institutes of Health (NIH) to start the Minority High School Research Apprenticeship Program.
6	Contradiction	In fact, regardless of the cost escalations on those other projects, legislators are doing their job by showing such prudence here.	when all is said and done, the legislature should approve the project aimed at repairing and upgrading seats and improving lighting and drainage at the facility.
7	Entailment	Also, this is the root of consciousness, because consciousness, awareness needs an opposite, a counterpart, a border, to awake at.	consciousness is a form of pain, originally, definitely.
8	Contradiction	Does Amaryl cause hair loss?	the use of Amaryl does not cause hair loss.
9	Entailment	"Golay has got no serious issues.	he is resorting to such statements for cheap popularity, " remarked Bhim Dahal, the spokesperson of ruling Sikkim Democratic Front (SDF).
10	Neutral	The point is, it has to go.	I'll be removing a lot of the buttons in favor of textual links, and will probably replace them with a single button promoting Firefox.
11	Contradiction	Most of the times precisely originating from a sincere analysis of the weaknesses, the tension field between opposites and the assembling of cross-functional teams, through the clash of diverse approaches and views, the influence of "career changers " from other fields, and openness towards the new, the unorthodox, the unpredictable.	without diminishing the importance of diversity, togetherness is what produces the most overwhelming feeling of success.
12	Entailment	Beef and poultry safety tips are essential to follow.	an effective and healthy way to lose weight is to get regular exercise and harmful toxins, as soon as you have to eat properly.
13	Neutral	One vehicle that is widely chosen is a motorbike to carry out daily activities.	the matic motor is very easy to operate.
14	Entailment	Storage companies are often located near major travel routes to make it easier for customers to access the facility.	you may see signs for local rental companies in your area at the side of the motorway, or major routes near your location.
15	Entailment	Connecting with people about a negative experience often equates to a positive outcome.	I ' ve decided to list all the ' abnormal ' things I do but wouldn ' t usually talk about.
16	Entailment	A smile does go a long way!	March 13, coming soon.
17	Contradiction	Accordingly, it is a cultural taboo to affirm, "I am Love, " which is our Authentic Self, the Immanent Divine Essence that we all share.	the Rishis who wrote the Upanishads realized that Brahman and Atman — as the Absolute and Self, respectively — are One, declaring Tat tvam asi ' You are That.
18	Contradiction	In these countries, ceramic proppants are used mainly in wells with higher closure pressures and other challenging environments.	ceramic proppants are the leading proppant type in the Chinese and Russian markets.
19	Contradiction	Given her recent prognosis and the fact that she DOES drive us mental with her meltdown and seemingly erratic behaviour sometimes, I really need to count to 10 and not lose my rag with her more than I do.	with her reaction to breakfast this morning and subsequent meltdown, I don't think I could have chosen a more difficult resolution...
20	Entailment	The Best Khao Soi in Chiang Mai, Thailand, is in a Mall!	I said it ... the best Khao Soi in Chiang Mai, Thailand, is in Central Airport Plaza Mall.

Table 18: Random sample of mined examples for NLI.