`ePiC`: Employing Proverbs in Context as a Benchmark for Abstract Language Understanding

Sayan Ghosh   Shashank Srivastava
UNC Chapel Hill
{sayghosh,ssrivastava}@cs.unc.edu

Abstract

While large language models have shown exciting progress on several NLP benchmarks, evaluating their ability for complex analogical reasoning remains under-explored. Here, we introduce a high-quality crowdsourced dataset of narratives for employing proverbs in context as a benchmark for abstract language understanding. The dataset provides fine-grained annotation of aligned spans between proverbs and narratives, and contains minimal lexical overlaps between narratives and proverbs, ensuring that models need to go beyond surface-level reasoning to succeed. We explore three tasks: (1) proverb recommendation and alignment prediction, (2) narrative generation for a given proverb and topic, and (3) identifying narratives with similar motifs. Our experiments show that neural language models struggle on these tasks compared to humans, and these tasks pose multiple learning challenges.

1 Introduction

Large language models (LLMs) (Devlin et al., 2019; Liu et al., 2019a; Raffel et al., 2020; Lewis et al., 2020; Reimers and Gurevych, 2019; Sanh et al., 2019; Lan et al., 2020) have led to a paradigm shift in NLP, and have shown exciting progress on benchmarks such as GLUE (Wang et al., 2019b) and SuperGLUE (Wang et al., 2019a). In particular, these include tasks such as reading comprehension, natural language inference, and coreference resolution. Many of these tasks rely on semantic and syntactic reasoning, which has been mastered by these LLMs. For example, apart from improving on distributional semantics through contextualized embeddings (Ethayarajh, 2019), recent work has shown evidence that these models implicitly learn emergent concepts such as subject-verb agreement (Jawahar et al., 2019), semantic roles (Tenney et al., 2019) and dependency structures (Hewitt and Manning, 2019).
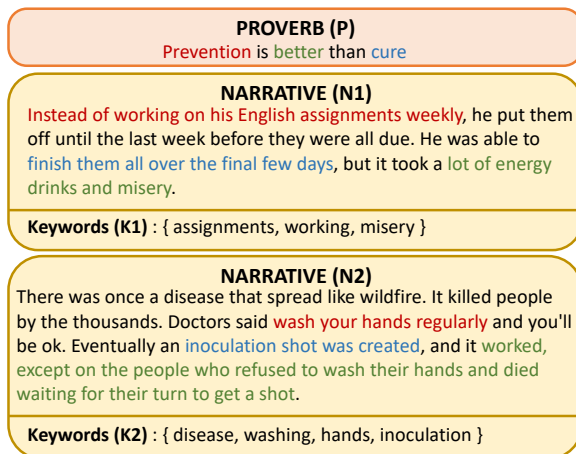

Figure 1: We introduce `ePiC`, a crowdsourced dataset of narratives for employing proverbs in context. Our dataset contains narratives (N1 and N2) paired against proverbs (P) along with a fine-grained annotation of *aligned spans* between the narratives and proverbs. Aligned spans are shown with matching colors and indicate correspondences in roles between proverbs and narratives. We explore three tasks: (1) proverb recommendation and alignment prediction (predict P given N1), (2) narrative generation for a given proverb and topic (generate N1 given P and K1), and (3) identifying narratives with similar motifs (e.g. identify N2 in a set of narratives given N1).

However, humans show an ability for deeper linguistic reasoning. We can identify people's intentions and goals (Douglas and Sutton, 2006), perform relational reasoning (Alexander et al., 2016), and find analogies in situations with little surface overlap (Holyoak, 2013). In particular, making verbal analogies in the form of proverbs is noted as an indicator of literary ability (Penfield and Duru, 1988; Nippold et al., 2001). Proverbs are also repositories of information on culture, societal norms, values, and folk wisdom (Raymond, 1956; White, 1987). In this work, we investigate proverbs in narrative contexts as a testbed for evaluating abstract reasoning and analogical abilities of LLMs.

We introduce `ePiC` (employing Proverbs in

3989

Context), a high-quality crowdsourced dataset of narratives paired with proverbs. The dataset provides fine-grained annotation of aligned spans between proverbs and narratives, and is designed to minimize lexical overlap between narratives and proverbs. Figure 1 shows two examples of narratives for a proverb from our dataset, along with corresponding alignment annotations. We diverge from related extant resources (Wang et al., 2020; Tan et al., 2015, 2016) on using proverbs in terms of quality of narratives, direct supervision, and having fine-grained alignment annotations.[1] We explore three tasks: (1) proverb and alignment prediction (§ 5.1), (2) narrative generation for a given proverb and a set of keywords specifying a topic (§ 5.2), and (3) discovering narratives with similar motifs (§ 5.3). By benchmarking several LLMs, we find that existing models struggle with these tasks, suggesting much scope of improvement in abstract reasoning. In particular, humans show much higher performance in many cases.

In §3, we describe the crowdsourced creation of the ePiC dataset. In §4, we analyze lexical overlap, biases, and narrative quality in ePiC. §5 describes the three tasks and details of experimental evaluation of LLMs for each task. We conclude with a discussion, and a statement of ethics and broader impact relevant to our work. Our contributions are:

- We introduce ePiC, a high-quality dataset for employing proverbs in context. It contains multiple narratives for English proverbs and fine-grained annotation of aligned spans between them.
- We design three challenging tasks that require models to go beyond surface-level reasoning and provoke research towards making more socially grounded NLP systems.
- We benchmark the performance of several state-of-the-art large language models in our proposed tasks using our dataset.

Our dataset and code are publicly available at: https://epic-benchmark.github.io

2 Related Work

Prior works in figurative language understanding have explored a diverse set of topics, such as simile detection and generation (Niculae and Danescu-Niculescu-Mizil, 2014; Mpouli, 2017; Zeng et al., 2020; Chakrabarty et al., 2020), metaphor detection

and generation (Dagan et al., 2005; Gao et al., 2018; Stowe et al., 2019, 2021; Chakrabarty et al., 2021b), pun identification (Poliak et al., 2018; Miller and Turković, 2016), and quote/proverb recommendation (Tan et al., 2015, 2016; Wang et al., 2020). Recent work (Chakrabarty et al., 2021a) has also focused on interpreting idioms and similes in narratives. Liu et al. (2019b) has explored recommending Chinese idioms through context-based recommendation and Zheng et al. (2019) formulated idiom recommendation as cloze-style reading comprehension task. Learning to quote has been explored based on fiction (Tan et al., 2015, 2016) and noisy social media conversations from Twitter, Reddit or Weibo (Lee et al., 2016; Wang et al., 2020). In the most related prior work, authors explore a quote retrieval task borrowing inspiration from context based recommendation systems (Huang et al., 2012; He et al., 2010). Wang et al. (2020) formulated learning to quote as a generation task by using topic modeling (Miao et al., 2017; Wang et al., 2019c) in a sequence-to-sequence network. While previous work has considered idioms, proverbs and common phrases as quotes, we specifically work with proverbs. Compared to earlier datasets, our dataset is manually created and labeled. Further, ePiC includes fine-grained annotations aligning parts of proverb to parts of the narrative, which has significant possibilities for model training, evaluation and interpretability.

3 Dataset Creation

In this section, we describe the steps involved in creating the dataset in detail.

Proverb collection: We obtained a candidate set of English proverbs by scraping websites of 'The Phrase Finder'[2] and WikiQuotes[3]. Next, this set was manually pruned to remove lexical variations of the same proverb. This manual curation led to a set of 250 proverbs, which we consider in the current version of our dataset.

Narrative collection: In the second step, we use Amazon Mechanical Turk to collect a diverse set of narratives corresponding to each proverb. We collect 10 narratives contributed by distinct turkers for each proverb, leading to a total of 2500 proverb-narrative pairs. We also ensure that no turker contributes a large number of narratives to alleviate

---

[1]Existing datasets are automatically created by scraping web-text, and supervision is heuristic (based on co-occurrences of proverbs and contexts)

[2]https://www.phrases.org.uk/
[3]https://en.wikiquotes.org/wiki/English_proverbs

annotator bias (Geva et al., 2019) (where models can overfit to annotator characteristics) while encouraging diversity in writing style and content. The turkers were asked to write short realistic stories, preferably within 100 words. Additionally, to avoid surface-form biases, turkers were encouraged to minimize lexical overlap and to not mention the proverb or parts of it in the narrative. This was done so that doing well on the tasks requires a detailed understanding of the narratives rather than relying on surface-level cues. Turkers were paid 50 cents for each narrative for this task.

Span alignment annotation: Next, we solicit fine-grained annotations between the narratives and the proverb in form of aligned spans. For this, we present proverb-narrative pairs to turkers asking them to find contiguous spans in the narrative which align well with contiguous spans in the proverb. Turkers could submit up to 5 pairs of aligned spans per proverb-narrative pair. These aligned spans highlight the grounding of a proverb in the narrative (see Figure 1). These annotations can help to verify the reasoning capabilities of various neural models by checking if these models are able to identify these correspondences, and add interpretability to our tasks. Turkers were paid 25 cents for each proverb-narrative pair annotation for this task.

Statistics: Table 1 shows the statistics of narrative collection for the proverbs. The narrative writing task was perceived as challenging yet interesting by most turkers due to (a) not having outlines about topics for the narrative beforehand (b) requirement of low lexical overlap with the proverb. Thus, the narrative writing task had a learning curve and some of the narratives submitted initially were not included in the dataset.

# submitted narratives	2561
# approved narratives	2500
# workers participated	166
Avg. # approved narratives per turker	15.06
Max # approved narratives by one turker	168

Table 1: Statistics of AMT task for narrative collection.

4 Dataset Analysis

Table 2 shows some statistics of the dataset collected through the process described in §3. In this section, we analyze the characteristics and biases of the ePiC dataset in detail.

Vocabulary size	16170
Avg. no. of tokens per narrative	64.27
Avg. no. of sentences per narrative	4.26
Avg. no. of aligned spans	2.18
Avg. no. words per proverb span	2.71
Avg. no. words per narrative span	11.57
No. of unique bigrams	80978
No. of unique trigrams	133772

Table 2: Dataset statistics for ePiC.

N-GRAM	JACCARD SIM.	COMMON N-GRAMS
unigram	0.0258 (0.0211)	1.27 (1.06)
bigram	0.0010 (0.0004)	0.07 (0.03)
trigram	0.0003 (0.0000)	0.02 (0.00)

Table 3: Avg. Jaccard similarity and number of common n-grams between proverbs and narratives. Numbers in parenthesis denote the corresponding statistics upon random assignment of proverbs to narratives.

4.1 Lexical overlap analysis

Using n-grams: We evaluate the extent of lexical overlap between proverbs and narratives by computing common n-grams between them. Table 3 reports the average Jaccard similarity score between n-gram sets of proverbs and narratives, and the average number of common n-grams. On average, there are 1.27 unigrams common between narratives and proverbs (including stopwords). In comparison, randomly permuting assignments of proverbs for narratives yields an average unigram Jaccard similarity of 0.0211 and 1.06 common unigrams. Thus, the overlap metrics in the dataset are comparable to those between unrelated texts.

To evaluate diversity among narratives corresponding to a proverb, we compute average Jaccard similarity between sets of unigrams for the narratives. This score is 0.107, which is comparable to a value of 0.098 for unigram overlap between pairs of narratives from different proverbs. This suggests a high lexical diversity between narratives.

Using distributional embeddings: We explore if we can retrieve the correct proverb corresponding to a narrative only by using similarity in their distributional representations. The similarity between a proverb and a narrative is defined as the cosine similarity between the representation of the proverb and the narrative obtained using word2vec embeddings (Mikolov et al., 2013) or contextual embeddings from LLMs. Details of implementation are provided in Appendix §F.1.

For this retrieval task, we report the accuracy and Mean Reciprocal Rank of the correct proverb in

LLM	ACC. (%) ↑	MRR ↑
Random	0.40	0.024
Word2Vec	1.52	0.047
BERT	0.36	0.025
ROBERTA	1.64	0.054
DistilBERT	1.92	0.053
ALBERT	0.40	0.025
Sentence-BERT	13.44	0.217
GPT-2	0.92	0.033
BART	1.14	0.041
T5	2.32	0.065

Table 4: Proverb retrieval performance using word2vec and off-the-shelf LLMs ('base' versions).

Table 4. We note that while all models perform better than random (with Sentence-BERT performing the best), the performance is very low when using out-of-the-box representations. In §5, we explore learning-based methods for the same setup.

4.2 Data characteristics

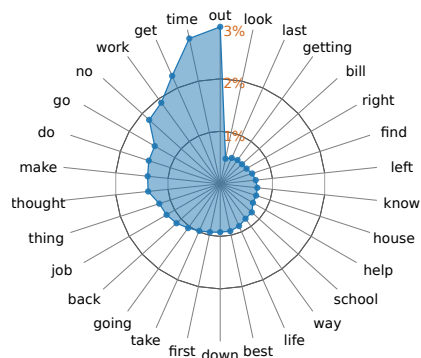**Diversity of narrative events:** Fig 2 shows the

Figure 2: Top-30 'event'/'process' hyponyms in ePiC.

distribution of events in our dataset. Following Mostafazadeh et al. (2016) we find events as the hyponyms of the word 'event' or 'process' using WordNet (Fellbaum, 2010). We see that the top events comprise less than 3% of all events in our dataset, and the long tail of less frequent events shows the diversity of the dataset.

Sentiment analysis: To evaluate the presence of sentiment association bias between proverbs and corresponding narratives (e.g., if negative sentiment proverbs only correspond to negative sentiments in narratives), we perform sentiment analysis of the narratives using VADER (Hutto and Gilbert, 2014). Figure 3 shows the average sentiment scores of the narratives corresponding to a proverb plotted against the sentiment score of the proverb. We find that the narratives are diverse in terms of their senti-

CRITERION	ePiC	[1]	[2]
Relatedness	3.91	3.15	**3.92**
Interesting/Creative	3.57	3.34	**3.63**
Fluency	**3.98**	3.23	3.80
Overall	**3.68**	3.15	3.66

Table 5: Averaged Likert scale ratings for data quality. Overall ratings for ePiC are better than [1] Wang et al. (2020) and [2] Tan et al. (2015).

ment polarities showing a weak positive correlation (Pearson correlation score 0.35) with the sentiment score of the proverbs. Figure 4 shows the variance
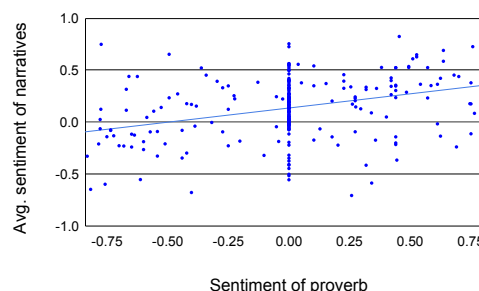

Figure 3: Average VADER sentiment score of narratives corresponding to a proverb against the VADER sentiment score of the proverb. The blue line shows the least-squares fit.

in terms of the number of positive and negative sentiment narratives (out of 10) for each proverb, showing a diverse spread of narrative sentiment polarities across proverbs. For additional details, please refer to Appendix §A.

We perform a few additional analyses on our dataset and found that (1) around 61% of mentions in the narratives were male, (2) diverse spread of reading complexity values in narratives measured using *Fleisch reading ease*[4], and (3) absence of any hate speech in the narratives of our dataset. The detailed experiments for these analyses are given in Appendix §A.

4.3 Human Evaluation of Dataset Quality

We perform a human evaluation of the narratives in our dataset on various criteria to judge the quality of our dataset. We perform this evaluation using the AMT platform. We randomly sample 250 proverb-narrative pairs and ask the turkers to evaluate the narratives on the following criteria:

• **Relatedness:** how closely the narrative reflects

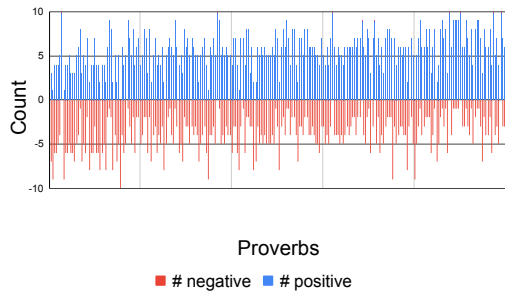[4] https://en.wikipedia.org/wiki/Flesch_Kincaid_readability_tests

Figure 4: Count of narratives with positive or negative VADER sentiment for each proverb. Proverbs are arranged in increasing order of their own VADER sentiment scores. Neutral sentiment narratives are excluded. For count of negative sentiment narratives (shown in red), consider the absolute value.

the meaning of the proverb (1: totally unrelated, 5: perfectly related)

- **Interesting/Creative:** how much is the narrative like a short creative or interesting story (1: very uninteresting/boring, 5: very creative/story-like)
- **Fluency:** grammatical correctness of the narrative (1: poor English with grammatical mistakes, 5: perfect English with no errors in writing)
- **Overall rating**

All the ratings are done on Likert scales from 1 to 5, where 1 is the lowest value for each criterion and 5 is the highest. Also, the rating value '3' was calibrated to be slightly leaning to the higher end of the scale (instead of neutral) so that the turkers take a clear stand on the polarity of each criterion. Table 5 shows the qualitative evaluation of our dataset. The average overall rating was 3.67 and the average pair-wise inter-annotator agreement for labeling a narrative as overall good vs overall poor (overall score >= 3 vs < 3) is 0.84[5]. We also rate the quality of the aligned spans in our dataset similarly on a scale of 1 to 5. The average rating of the alignment between spans was 3.91 and the average pair-wise inter-annotator agreement for alignment as good vs poor (rating >= 3 vs < 3) is 0.86[5].

Table 6 highlights the key differences between ePiC and prior work that dealt with related figurative language tasks involving quotes. Notably, ePiC exclusively deals with proverbs unlike prior work (which includes common phrases and idioms such as "trust your gut") and also provides granular annotations in form of annotated spans. Also,

---

ePiC contains narratives crowdsourced by specifically keeping proverbs in focus, rather than obtaining them using heuristic supervision. To quantify dataset quality, we ran human evaluation similar to ePiC over (1) 200 randomly drawn samples from the "Reddit" dataset of quotations in context from the Wang et al. (2020), and (2) 200 randomly drawn samples from the corpus of Tan et al. (2015). Based on average Likert scores in Table 5 we find that ePiC is (1) significantly superior (using t-test; $p < 0.05$) on all criteria than Wang et al. (2020), and (2) better in overall ratings than Tan et al. (2015).

5 Tasks & Evaluation

In this section, we introduce three tasks associated with ePiC and describe their experimental setup and benchmark results: (1) Proverb and Alignment Prediction, (2) Narrative Generation, and (3) Identifying narratives with similar motifs.

5.1 Proverb and alignment prediction

5.1.1 Task details

In this task, the objective is to predict the correct proverb for a given narrative from the set of 250 proverbs in the dataset. The motivation of this task is to test whether language models can abstract the underlying meaning of the narratives and make an analogy with the correct proverb from a large set of proverbs. In terms of applications, this task is related to proverb recommendation, which can be useful in creative writing assistants. The task is challenging as there might be multiple proverbs loosely related to the narrative context, but not be completely consonant with subliminal themes in the narrative. An underlying assumption here is that a narrative would match well with exactly one proverb. We found this reasonable for most examples in the dataset.

5.1.2 Experiment Setup and Results

We consider two settings, predicting (1) Seen and (2) Unseen proverbs.

- Seen proverbs: The set of proverbs in the train and test set are the same. We divide narratives corresponding to each proverb into train and test in 6:4 ratio. So, the train and test sets have 1500 and 1000 proverb-narrative pairs respectively.
- Unseen proverbs: Here, we consider 150 proverbs in the train set and the remaining 100 proverbs in the test set (6:4 split on the set of proverbs). The sets of proverbs in the train and

CHARACTERISTICS	Tan et al. (2015)	Lee et al. (2016)	Wang et al. (2020)	ePiC
Domain	Fiction	Social Media	Social Media	Fiction
Manual curation of narratives	✗	✗	✗	✓
Alignment annotation	✗	✗	✗	✓
Focus on proverbs	✗	-[6]	✗	✓

Table 6: Comparing `ePiC` with prior works on learning to quote based on different characteristics of the data and the collection process. While previous methods collect contexts and labels by mining existing text resources through heuristics (with no manual curation), `ePiC` contains contexts in form of narratives authored by crowdworkers explicitly for this task. `ePiC` further provides fine-grained alignment annotation between narratives and proverbs.

test split are disjoint. So, the train and test sets have 1500 and 1000 proverb-narrative pairs respectively (since each proverb has 10 narratives).

Proverb prediction: Here we focus on only predicting the corresponding proverb for a narrative, without employing the span alignments in training or evaluation. For this, we fine-tune the retrieval models based on different LLMs previously described in §4 (details of models in Appendix §F.2). To evaluate performance we consider accuracy and Mean Reciprocal Rank as metrics. Table 7 shows best proverb prediction performance on test split for 'seen' and 'unseen' proverbs[7]. RoBERTa performs the best for both the 'seen' and 'unseen' settings, and the performance for all models is consistently lower for unseen proverbs (as would be expected, since this task involves much greater generalization). Further, while the performance of all models is much better than chance, even the highest performance is only 28.2%.

MODEL	ACC. (%) ↑	MRR ↑
Seen proverbs		
Random	0.4	0.024
BERT	22.9	0.342
RoBERTa	28.2	0.391
DistilBERT	18.7	0.289
ALBERT	13.4	0.221
Sentence-BERT	20.6	0.315
BART	15.8	0.245
T5	18.7	0.292
Unseen proverbs		
Random	1.0	0.005
BERT	19.2	0.307
RoBERTa	20.3	0.314
DistilBERT	17.4	0.277
ALBERT	1.1	0.053
Sentence-BERT	17.0	0.278
BART	8.5	0.189
T5	13.7	0.242

Table 7: Proverb prediction performance on 'seen' and 'unseen' proverbs (all LLMs are in 'base' version).

Alignment prediction: Here we focus only on predicting an aligned span from the narrative given the narrative, proverb, and a span from the proverb as inputs. We fine-tune two large language models (BERT and RoBERTa) for this by adopting a learning framework similar to answer span prediction for SQUAD (Rajpurkar et al., 2016). The language model outputs two probability distributions corresponding to the start and end positions of a span, over the narrative tokens. We iterate over all the combinations of the start and end tokens and choose the span with maximum likelihood. For span prediction, we report token-level precision, recall, and F1. Table 8 shows the results of alignment

MODEL	SPAN P	SPAN R	SPAN F1
BERT	0.070	0.123	0.089
RoBERTa	0.068	0.143	0.092

Table 8: Alignment prediction performance for seen proverbs using LLMs ('base' versions).

prediction on the 'seen' proverbs using BERT and RoBERTa models. We find that the performance is low for both models indicating major scope for improvements.

Predicting proverbs and alignment jointly: We formulate this as multi-task learning. We extend the models from the proverb prediction task by adding a component to predict span from narrative given a span from the proverb and the narrative. The language model is thus shared across the proverb prediction and span prediction tasks. The span prediction branch predicts the start and end position of the corresponding narrative span. We jointly train the model with multi-task learning of the two tasks, i.e., proverb and alignment prediction, on the 'seen' proverbs data split. We report the accuracy for proverb prediction and precision, recall, and F1 for span prediction. Apart from this joint model, we also consider a pipelined baseline model which first does proverb prediction, followed by span pre-

---

[7]Our reported accuracies denote the highest accuracy achieved on the test set during model training as we do not have a validation set to choose the best model.

[6]We did not have access to the dataset to verify this.

diction if the correct proverb was predicted. Table 9 shows results for the joint model and the pipelined-baseline model. The low performance of the models indicates major scope for improvements in the individual tasks. While in principle the two tasks should benefit from joint training, we find that joint training performs worse than pipelined-baseline for both proverb and alignment prediction. Future work can explore designing better models for joint training to leverage the interdependence between proverb prediction and alignment prediction.

MODEL	ACC. (%)	SPAN P	SPAN R	SPAN F1
Pipelined				
BERT	22.9	0.018	0.035	0.024
RoBERTa	28.2	0.019	0.048	0.027
Joint training				
BERT	19.8	0.015	0.029	0.019
RoBERTa	26.5	0.015	0.030	0.020

Table 9: Joint proverb and alignment prediction performance for seen proverbs using LLMs ('base' versions).

5.1.3 Qualitative analysis of proverb prediction models

Figure 5 shows a heatmap to study the differences in prediction accuracies of BERT and RoBERTa models. We see that RoBERTa generally outperforms BERT for many cases (in Figure 5, values in the bottom-right triangle are typically greater than the top-left). Looking into the narratives for proverbs in the test set with high accuracy (>=0.75), we think a reason for the high performance could be the presence of certain words/phrases which are synonymous to some words/phrases in the proverb (for example, presence of word 'group' for the proverb '*birds of a feather flock together*'). On the other hand, there are cases when the model is confused because of multiple topics being discussed in the narrative resulting in an incorrect prediction. For example, some narratives in the test set for the proverb '*life's not all beer and skittles*' describe earning money the hard way, which confused the RoBERTa model into predicting '*time is money*' for such narratives.

5.1.4 MCQ task for human performance comparison

To formulate a feasible task for humans, we frame proverb prediction as a multiple choice question (MCQ) task where for each narrative, 5 proverbs are provided as choices. The set of choices includes the correct proverb and 4 other distractor

Figure 5: Heatmap showing the percentage of proverbs with various fine-tuned BERT and RoBERTa proverb prediction accuracies (for example, more than 15% of the proverbs have RoBERTA prediction accuracy as 25% and BERT prediction accuracy as 25%).

proverbs, chosen by using the fine-tuned RoBERTa model. Examples of the MCQ task and details of choosing distractors are provided in Appendix §B. Table 10 shows the accuracy of the human evaluation for this MCQ task on a random sample of 100 narratives from the test split of "seen" proverbs conducted using AMT. Compared to RoBERTa, we find humans are much better at this adversarially created MCQ task. Note that the performance for RoBERTa in Table 10 and Table 7 is different, as Table 10 reports accuracy only on the random sample of narratives chosen for human evaluation. The estimate for human performance is likely an under-estimate since in many cases human subjects were unfamiliar with the meanings of some of the proverbs provided in the options and as a result, focused more on surface-level cues (details of this analysis are provided in Appendix §B). The average pair-wise inter-annotator agreement between human subjects for this task was 0.73[5].

This evaluation does not take into account semantic similarity between proverbs (two proverbs might be equally suitable for the same context). To explore this, we analyze the human errors on the MCQ task and find that in only around 11% of the errors, the proverb chosen by humans is semantically similar to the annotated proverb and can also be a suitable answer to the MCQ task. Details about this analysis are given in Appendix §C. Future work can consider handling semantic similarity between proverbs explicitly and devise suitable evaluation metrics.

Predictor	ACC.(%) ↑
RoBERTA	23.0
Human	78.7

Table 10: Proverb prediction accuracy in MCQ setting.

5.2 Narrative Generation

5.2.1 Task details

One of the important use-cases for NLP models in the creative writing domain is to use these models to generate content. We explore the task of generating narratives corresponding to a proverb and a given topic (specified as a set of keywords). We benchmark the performance of two recently proposed state-of-the-art models in text generation, T5 (Raffel et al., 2020) and BART (Lewis et al., 2020), by fine-tuning them on ePiC.

5.2.2 Experiments and Results

We divide our dataset into train and test split under 'seen' and 'unseen' proverbs settings similar to the proverb prediction task. We consider the set of verbs and named-entities as the keywords for a narrative. We train our narrative generation model conditioned on the proverb and the keywords.

Table 11 shows results for automatic evaluation of the generated narratives using BLEU (Papineni et al., 2002), ROUGE-L (Lin, 2004), and recall of the keywords mentioned in the generated narrative as metrics. Examples of generated narratives are given in Appendix §D. We find that BART performs better than T5 on the automatic evaluation metrics. Further, we perform human evaluation to evaluate the quality of the generated narratives in AMT by considering the same criteria (and rating semantics) employed in Section 4.3. Table 12 shows the human evaluation of generated narratives using BART and T5 when tested over 'seen' proverbs. Low scores for BLEU and ROUGE-L in automatic metrics and low Likert ratings of the generated narratives indicate much scope for future improvement on this task.

5.3 Identifying narratives with similar motifs

5.3.1 Task details

An important aspect of language understanding is the ability to make linguistic (and narrative) analogies, i.e., identifying 'similarity' between narratives (e.g., identifying two narratives that are variations on the 'Cinderella story' theme). Here, we explore the task of identifying narrative analogy by modeling 'similarity' between narratives based

MODEL	BLEU ↑	ROUGE-L ↑	RECALL ↑
Seen proverbs			
BART	4.21	30.80	0.90
T5	2.25	27.83	0.77
Unseen proverbs			
BART	4.39	31.36	0.93
T5	2.34	26.61	0.75

Table 11: Automatic evaluation for narrative generation on 'seen' and 'unseen' proverbs using 'base' versions of LLMs.

CRITERION	BART	T5
Relatedness	2.75	2.57
Interesting/Creative	2.97	3.07
Fluency	2.71	2.53
Overall	2.87	2.76

Table 12: Human evaluation results for narrative generation on 'seen' proverbs.

on proverbs illustrated by them. For this task, two narratives are taken to be similar if they are related to the same proverb.

5.3.2 Experiments and Results

For this task, we use the train and test split of 'seen' proverbs setup in the proverb prediction task. The aim is to find similar narratives for each narrative in the test split amongst all narratives in the test split. So for each narrative, there are 3 other similar narratives (corresponding to the same proverb) in the test split (containing 1000 narratives).

Modeling similarity between narratives We use the learned models in the proverb prediction task to obtain a probability distribution over the proverbs for each narrative. To model similarity, we compute the distance between the (vectors representing) two probability distributions using one of the following: (1) cosine distance; (2) Jenson-Shannon divergence; (3) L2 (Euclidean) distance; and (4) L1 (Manhattan) distance. We predict the narrative closest (in terms of distance metrics) to the input narrative as the most similar. Table 13 shows the accuracy of getting a similar narrative using different distance metrics and different fine-tuned LLMs. Using cosine or Jenson-Shannon divergence as the distance metric on the probability distribution over proverbs predicted by the RoBERTa model performs best on this task. However, the overall performance of models are still low and can be benefited by devising suitable training methods for this task.

We perform an additional experiment on finding similar narratives without performing proverb prediction as an intermediate step. We use a pretrained Sentence-BERT model to obtain representations of each narrative. For a given input narrative, we calculate the cosine distance between the Sentence-BERT representation of the input narrative and all other narratives in the test set. We predict the narrative having minimum cosine distance to the input narrative as the most similar. Using this approach we find the accuracy of identifying similar narratives as 6.6%, which is lower than most values reported in Table 13. This low value highlights the diversity between narratives and the challenge in finding analogies between narratives.

MODEL	Cos	JSD	L2	L1
BERT	8.5	8.0	7.3	7.9
RoBERTa	13.3	13.4	11.2	11.8
Distil-BERT	6.5	7.2	5.2	6.0
Sentence-BERT	7.2	6.1	7.0	5.9

Table 13: Prediction accuracy (%) for identifying similar narratives by using different distance metrics and distribution over proverbs from different LLMs ('base' versions).

6 Conclusion and Future Work

We introduce ePiC, a high-quality crowdsourced dataset of narratives paired with proverbs, and a suite of challenging tasks associated with this dataset. We show that these provide a challenging testbed for evaluating abstract reasoning and analogical abilities of LLMs. Future work can explore more sophisticated mechanisms to use alignment annotations in improving the performance for proverb prediction and model interpretability. Additionally, researchers can explore conditional narrative generation through more informative prompts than using keywords. ePiC can also be extended in the future by incorporating more proverbs and adding more layers of complexity like sarcasm or adversarially creating harder narratives. Most of all, the development of similarly challenging resources and tasks can enable the possibility of socially grounded NLP systems.

Ethics and Broader Impact

In §4, we note that our dataset shows considerable differences in the distribution of gender of entities (61% male vs 39% female), whereas in the real world we expect the ratios to be about equally balanced. Systems that don't account for this bias might end up performing better for narratives with male entities than with females. However, we note that narratives with male and female entities show no differences in overall length or the average number of mentions to those entities.

The proverbs used in our dataset were collected from free public resources without violating intellectual property rights. We do not collect any personal information from the turkers who participated in our crowdsourced tasks. We release our dataset publicly without mentioning any personal details of turkers available automatically in AMT (such as turker IDs). The turkers were compensated fairly and the payment per task is equivalent to an hourly compensation that is greater than minimum wage (based on the median time taken by turkers).

For all the crowdsourcing tasks in this work, we limited the locale of eligible turkers to the USA, Canada, and the UK. Further, to ensure good-faith turkers, we required that the approval rate of the turkers be above 97%.

Our screening process has selection biases that likely over-samples narrative-writers from demographics that are over-represented on AMT (ethnically white, college-educated, lower-to-medium income, and young) (Hitlin, 2016), and this is likely to have affected the topics and type of language usage in the collected narratives.

Finally, our investigation here has focused on traditional English proverbs, even while proverbs are universal in human languages and cultures (Penfield and Duru, 1988). This poses a real risk of the development of AI models that better understand and employ specific types of figurative language than others. Such systems are likely to be less user-friendly to users that don't belong to specific social-cultural backgrounds. To mitigate these risks, but also since proverbs are universal repositories of culture-specific knowledge, future work should extend our effort to more equitably represent the variety and diversity of human thought and cultural experiences. Our investigation here, unfortunately, does not adequately do this. As the proverb goes, the road to hell is paved with good intentions.

References

Patricia A Alexander, Denis Dumas, Emily M Grossnickle, Alexandra List, and Carla M Firetto. 2016.

Measuring relational reasoning. *The Journal of Experimental Education*, 84(1):119–151.

Tuhin Chakrabarty, Yejin Choi, and Vered Shwartz. 2021a. It's not rocket science: Interpreting figurative language in narratives. *arXiv preprint arXiv:2109.00087*.

Tuhin Chakrabarty, Smaranda Muresan, and Nanyun Peng. 2020. Generating similes effortlessly like a pro: A style transfer approach for simile generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6455–6469, Online. Association for Computational Linguistics.

Tuhin Chakrabarty, Xurui Zhang, Smaranda Muresan, and Nanyun Peng. 2021b. MERMAID: Metaphor generation with symbolism and discriminative decoding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4250–4261, Online. Association for Computational Linguistics.

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

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11.

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.

Karen M Douglas and Robbie M Sutton. 2006. When what you say about others says something about you: Language abstraction and inferences about describers' attitudes and goals. *Journal of Experimental Social Psychology*, 42(4):500–508.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Alvan R Feinstein and Domenic V Cicchetti. 1990. High agreement but low kappa: I. the problems of two paradoxes. *Journal of clinical epidemiology*, 43(6):543–549.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

Ge Gao, Eunsol Choi, Yejin Choi, and Luke Zettlemoyer. 2018. Neural metaphor detection in context. *arXiv preprint arXiv:1808.09653*.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China. Association for Computational Linguistics.

Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. 2010. Context-aware citation recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 421–430.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Paul Hitlin. 2016. 4. turkers in this canvassing: Young, well-educated and frequent users. *Pew Research Center*, 437.

Keith J Holyoak. 2013. Analogy and relational reasoning. *The Oxford Handbook of Thinking and Reasoning*, page 234.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spacy: Industrial-strength natural language processing in python.

Wenyi Huang, Saurabh Kataria, Cornelia Caragea, Prasenjit Mitra, C Lee Giles, and Lior Rokach. 2012. Recommending citations: translating papers into references. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1910–1914.

Clayton Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 8.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Hanbit Lee, Yeonchan Ahn, Haejun Lee, Seungdo Ha, and Sang-goo Lee. 2016. Quote recommendation in dialogue using deep neural network. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 957–960.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pretraining for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

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. *arXiv preprint arXiv:1907.11692*.

Yuanchao Liu, Bo Pang, and Bingquan Liu. 2019b. Neural-based Chinese idiom recommendation for enhancing elegance in essay writing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5522–5526, Florence, Italy. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. In *International Conference on Machine Learning*, pages 2410–2419. PMLR.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tristan Miller and Mladen Turković. 2016. Towards the automatic detection and identification of english puns. *The European Journal of Humour Research*, 4(1):59–75.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Suzanne Mpouli. 2017. Annotating similes in literary texts. In *Proceedings of the 13th Joint ISO-ACL Workshop on Interoperable Semantic Annotation (ISA-13)*.

Vlad Niculae and Cristian Danescu-Niculescu-Mizil. 2014. Brighter than gold: Figurative language in user generated comparisons. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 2008–2018.

Marilyn A Nippold, Melissa M Allen, and Dixon I Kirsch. 2001. Proverb comprehension as a function of reading proficiency in preadolescents.

Travis E Oliphant. 2006. *A guide to NumPy*, volume 1. Trelgol Publishing USA.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Joyce Penfield and Mary Duru. 1988. Proverbs: Metaphors that teach. *Anthropological quarterly*, pages 119–128.

Adam Poliak, Aparajita Haldar, Rachel Rudinger, J Edward Hu, Ellie Pavlick, Aaron Steven White, and Benjamin Van Durme. 2018. Collecting diverse natural language inference problems for sentence representation evaluation. *arXiv preprint arXiv:1804.08207*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

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.

Joseph Raymond. 1956. Tensions in proverbs: more light on international understanding. *Western Folklore*, 15(3):153–158.

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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. 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*.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. In *International Conference on Machine Learning*, pages 4596–4604. PMLR.

Kevin Stowe, Tuhin Chakrabarty, Nanyun Peng, Smaranda Muresan, and Iryna Gurevych. 2021. Metaphor generation with conceptual mappings. 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 6724–6736, Online. Association for Computational Linguistics.

Kevin Stowe, Sarah Moeller, Laura Michaelis, and Martha Palmer. 2019. Linguistic analysis improves neural metaphor detection. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 362–371.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2015. Learning to recommend quotes for writing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2016. A neural network approach to quote recommendation in writings. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 65–74.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman,

Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019a. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019b. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

Lingzhi Wang, Jing Li, Xingshan Zeng, Haisong Zhang, and Kam-Fai Wong. 2020. Continuity of topic, interaction, and query: Learning to quote in online conversations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6640–6650, Online. Association for Computational Linguistics.

Yue Wang, Jing Li, Hou Pong Chan, Irwin King, Michael R. Lyu, and Shuming Shi. 2019c. Topic-aware neural keyphrase generation for social media language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2516–2526, Florence, Italy. Association for Computational Linguistics.

Geoffrey M White. 1987. Proverbs and cultural models: An american psychology of problem solving.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Jiali Zeng, Linfeng Song, Jinsong Su, Jun Xie, Wei Song, and Jiebo Luo. 2020. Neural simile recognition with cyclic multitask learning and local attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9515–9522.

Chujie Zheng, Minlie Huang, and Aixin Sun. 2019. ChID: A large-scale Chinese IDiom dataset for cloze test. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 778–787, Florence, Italy. Association for Computational Linguistics.

Appendix

A   Additional dataset analysis

Additional details on sentiment analysis: An example of proverb for which the narratives were close in sentiment scores to the proverb is '*a thing of beauty is a joy forever*' while for '*there's no fool like an old fool*' the sentiment polarity of narratives was on average opposite to that of the proverb. We note that there are indeed a small number of proverbs for which all or most narratives leaning towards a particular sentiment polarity. Quantitatively, for 23 proverbs, either 9 or all 10 of the narratives have positive VADER sentiment score. These include: '*Nothing succeeds like success*' , '*Christmas comes but once a year*' and '*Genius is one percent inspiration, ninety-nine percent perspiration*'. There are 6 proverbs for which either 9 or all 10 narratives have a negative VADER sentiment score. These include: '*The wages of sin is death*', '*Fish always stink from the head down*' and '*Don't wash your dirty linen in public*'. However, as seen in Figure 4, the vast majority of proverbs in the dataset are represented by narratives with both positive and negative sentiment polarities.

Gender distribution of entities: Using an off-the-shelf neural coreference pipeline, we find that 61% of the mentions in the narratives are male, while 39% are female. Around 48% of the narratives have predominantly male mentions, 26% of the narratives have predominantly female mentions and the rest have equal number of male and female mentions. The average number of words in predominantly male and female mention containing narratives was comparable (65 words).

Language complexity: We use the *Fleisch reading ease*[6] to calculate language complexity of narratives in our dataset. The reading scores vary from 112.1 (equivalent to 3rd grade reading levels) to -41.5 (significantly above college graduate reading levels) with an average score for the narratives in our dataset as 66.5 (equivalent to 8th/9th grade reading levels), showing a considerable spread in the complexity of language in our dataset.

Hate speech: Using an off-the-shelf hate speech classifier (Davidson et al., 2017), we found no instances of hate or toxic speech in the dataset.

---

[6]https://en.wikipedia.org/wiki/Flesch_Kincaid_readability_tests

B   Human evaluation on MCQ task

We formulated a MCQ task for proverb prediction to gauge human performance. The MCQ task has 5 options – correct proverb and 4 distractor proverbs. The distractor proverbs were chosen using the fine-tuned RoBERTa model on the proverb prediction task. We choose the distractor proverbs from a mix of proverbs with the highest prediction probabilities, and proverbs that are assigned the most similar probabilities to the correct answer from the RoBERTa model. We performed this study using the Amazon Mechanical Turk platform. We observed that this task is not that simple even for humans and requires a certain level of proficiency in English language or in proverbs specifically. The task is more challenging since the options other than the correct choice in the MCQ task were chosen by picking the most confusing options deemed by the RoBERTa (Liu et al., 2019a) model. However, we find that these wrong choices are confusing for humans too. This is because superficially these wrong choices also seem quite related to the narrative and it requires good reasoning skills to identify the correct narrative. The other situation where the turkers failed was when the options contained multiple proverbs which are quite close in meaning. For example, when the options contained both '*there's no accounting for tastes*' and '*Beauty is in the eye of the beholder*' the turkers often chose the former when the annotated proverb was the latter. Table 14 shows examples of narratives along with the choices of proverbs where turkers failed to identify the correct proverb.

C   Semantically similar proverbs

Our chosen set of 250 proverbs in ePiC includes instances of proverbs that are semantically very similar, or even paraphrases (e.g., '*never judge a book by its cover*' and '*appearances can be deceptive*'). This can be problematic since the presence of semantically similar proverbs as different options in MCQ (and as different classes in proverb classification task) can confuse both humans and automated models. To estimate the extent of this phenomenon, we perform an analysis of human errors on the aforementioned MCQ task. Out of 64 errors we find that for 20 cases, the chosen proverb was completely unrelated to the actual answer. For 29 out of the remaining 44 cases, the chosen proverb *seems* related to the narrative at first glance, but is not aligned and thus not the best fit. For the remaining

Narrative 1:	

Narrative 1:
She had been so happy when he had asked her to marry him but three years on, it seemed that he had so many excuses for not setting a date that she thought that it was never going to happen. Her happiness eventually turned to despair and she considered breaking the engagement.

Choice A : You win some, you lose some
Choice B : Jam tomorrow and jam yesterday, but never jam today (*Correct*)
Choice C : Cowards may die many times before their death
Choice D : The course of true love never did run smooth (*Marked*)
Choice E : Nothing is certain but death and taxes

Narrative 2:
She didn't want to embarrass her friend when she asked her, "It's beautiful, isn't it?" She looked at her friend's new car and nodded her head in agreement. It was purple, the worst car colour she had ever seen, but she faked a smile and congratulated her.
Choice A : Imitation is the sincerest form of flattery
Choice B : From the sublime to the ridiculous is only one step
Choice C : There's no accounting for tastes (*Marked*)
Choice D : Beauty is in the eye of the beholder (*Correct*)
Choice E : All publicity is good publicity

Table 14: Tricky MCQ questions from human evaluation task of proverb prediction: The above samples show the challenges in the human evaluation task. In case of narrative 1, the turkers often confuse with choice D which superficially seems related but is not correct. For narrative 2, the proverbs in choices C and D are quite close in meaning, thus resulting in a wrong choice by turkers.

15 cases (23% of human errors), the chosen proverb would have been equally appropriate for the narrative. Further, in 7 out these 15 cases (11% of human errors), the chosen proverb is also semantically similar to the annotated proverb. Future work can consider handling semantic similarity between proverbs explicitly and devise suitable evaluation metrics.

D Generated Narratives

We show some examples of the narratives generated by the BART (Lewis et al., 2020) and T5 (Raffel et al., 2020) models for the narrative generation task in Table 15. We see that even though the models try to mention all the keywords but they are not able to generate a coherent narrative.

E Evaluation of alignment prediction for jointly trained models

In §5.1, we present models to predict proverb and alignment jointly. During evaluation, we first perform proverb prediction and then perform alignment prediction if the correct proverb was pre-

dicted. If an incorrect proverb is predicted, the span precision, span recall, and span F1 are considered as zero.

If we isolate the layers responsible for alignment prediction from the joint model and evaluate it solely on the task of alignment prediction, the span F1 scores for BERT (base) and RoBERTa (base) are 0.078 and 0.074 respectively.

F Training details

In this section we discuss about the model parameters, hyper-parameter settings and hardware and software specifications of training.

F.1 Retrieval models' implementation details

As discussed in §4.1, we formulate a retrieval task to explore if we can retrieve the correct proverb corresponding to a narrative only by using similarity in their distributional representations. We define similarity between a proverb and a narrative by using cosine similarity between the embeddings of the proverb and the narrative. We use (1) word2vec embeddings (Mikolov et al., 2013) (2) contextual embeddings from LLMs to represent the proverb

Proverb 1:	

Proverb 1:
Absence makes the heart grow fonder
Gold narrative:
Although they never got along together much, she somehow always looked forward
to the holidays each year.
Keywords:
got, looked, the holidays
Generated narrative by BART:
Polly got back into the swing of things, she looked forward to the holidays each year.
Generated narrative by T5:
Although he got through the holidays each year, he looked over everything
again to his friends, family and friends.

Proverb 2:
Those who do not learn from history are doomed to repeat it
Gold narrative:
Bob cheated on his girlfriend in college. He was heartbroken when she broke up with
him but moved on. A few years later he married and cheated on his wife. She left him,
just as his girlfriend did in college.
Keywords:
broke, did, a few years later, cheated, moved, left, married, heartbroken, Bob
Generated narrative by BART:
Bob cheated on his girlfriend with her best friend. He was heartbroken when she broke
up with him and moved out of his apartment. A few years later,he got married to the
girl he cheated on with, and left his apartment with her. He never did again.
Generated narrative by T5:
Bob was heartbroken. a few years later, he broke into town. He moved out in his life and
moved out.. a few years later, he married a girl. A few years later, she cheated on him but,
unfortunately, she left him because she did not do well.

Proverb 3:
A trouble shared is a trouble halved
Gold narrative:
Mike knows that this new problem is going to need some serious help. He tells his friend
Josh about the new problem,and Josh agrees to help him solve it.
Keywords:
help, solve, knows, going, Mike, Josh, need, tells, agrees
Generated narrative by BART:
Mike knows a thing or two about what is going on in his life that he need help with
anything. he tells his friend josh about it and Josh agrees to help solve the problem.
Generated narrative by T5:
Mike, Josh, knows that he is going to need help to solve the problem. He tells me that
he agrees but he will not help me solve the problem.

Table 15: Examples of generated narratives by BART and T5 on 'seen' proverbs

and narrative. We obtain the embeddings for a context c (where c can be a proverb or a narrative) as:

- Word2vec: average of word embeddings for tokens in c.
- BERT (Devlin et al., 2019)/RoBERTa (Liu et al., 2019a) : $[CLS]$ token embedding on passing c through BERT/RoBERTa.
- DistilBERT (Sanh et al., 2019)/AlBERT (Lan et al., 2020) : $[CLS]$ token embedding on passing c through DistilBERT/AlBERT
- SentenceBERT (Reimers and Gurevych, 2019) :

normalized SentenceBERT embeddings obtained by using 'all-mpnet-base-v2'[7] model on c.

- T5 (Raffel et al., 2020)/GPT-2 (Radford et al., 2019) Encoder: sum of embeddings of tokens in c after passing through the encoder

F.2 Proverb prediction models' implementation details

We use the same LLM models (and implementations) used for the retrieval setup discussed in §4.1 and §F.1.

F.3 Obtaining keywords for narrative generation

We consider the named entities and verbs present in a narrative (extracted using `spacy` (Honnibal et al.)) as keywords for generating that narrative.

F.4 Model parameters

Our proverb prediction models do not introduce any additional parameters over the existing parameters in the large language models. For joint prediction of proverb and span, we introduce new fully connected layers over the language models, thus introducing 0.6 M additional parameters.

F.5 Hyper-parameter settings

For all the transformer based models we use the implementation of HuggingFace library (Wolf et al., 2019). All the model based hyper-parameters are thus kept default to the settings in the HuggingFace library. We use the publicly available checkpoints to initialise the pre-trained models (for example "bert-base-uncased" checkpoint for initialising BERT(Devlin et al., 2019)). For the proverb prediction models we did not truncate any tokens from the proverb and considered the maximum length of the narrative sequence to be 256 tokens. For the alignment prediction and joint training models, we considered the maximum length of the narrative sequence as 230 tokens. We used the AdamW (Loshchilov and Hutter, 2018) optimizer commonly used to train these models except for T5 (Raffel et al., 2020). We used AdaFactor(Shazeer and Stern, 2018) to train our T5 based proverb prediction model. We kept the learning rate as 0.00002 for training. Batch sizes was kept as 16 except for T5, for which we reduced the batch size to 4. The random seed for all experiments was 42. The proverb prediction models were trained for 25

epochs. The BART narrative generation model was trained for 15 epochs and loss converged after that. T5 took longer and was trained for 25 epochs.

F.6 Software and hardware specifications

All the models are coded using Pytorch 1.4.0[8] (Paszke et al., 2019) and related libraries like numpy (Oliphant, 2006), scipy (Virtanen et al., 2020) etc. We run all experiments on GeForce RTX 2080 GPU of size 12 GB. The system has 256 GB RAM and 40 CPU cores. The proverb prediction models typically take 2-5 mins for one epoch. For the joint proverb and span prediction models it took roughly 10 mins for one epoch. For narrative generation models it takes 10 mins for BART and around 18 mins for T5 to complete one epoch of training.

---

[7] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

[8] https://pytorch.org/