

Retrieving Texts based on Abstract Descriptions

Shauli Ravfogel^{1,2} Valentina Pyatkin^{1,2}

Amir DN Cohen¹ Avshalom Manevich¹ Yoav Goldberg^{1,2}


¹Bar-Ilan University ²Allen Institute for Artificial Intelligence

{shauli.ravfogel, valpyatkin, amirdnc, avshalomman, yoav.goldberg}@gmail.com

Abstract

In this work, we aim to connect two research areas: instruction models and retrieval-based models. While instruction-tuned Large Language Models (LLMs) excel at extracting information from text, they are not suitable for semantic retrieval. Similarity search over embedding vectors allows to index and query vectors, but the similarity reflected in the embedding is sub-optimal for many use cases.

We identify the task of retrieving sentences based on abstract descriptions of their content. We demonstrate the inadequacy of current text embeddings and propose an alternative model that significantly improves when used in standard nearest neighbor search. The model is trained using positive and negative pairs sourced through prompting an a large language model (LLM). While it is easy to source the training material from an LLM, the retrieval task cannot be performed by the LLM directly. This demonstrates that data from LLMs can be used not only for distilling more efficient specialized models than the original LLM, but also for creating new capabilities not immediately possible using the original model.

 <https://huggingface.co/biu-nlp/abstract-sim-sentence>
<https://huggingface.co/biu-nlp/abstract-sim-query>

1 Introduction

A common—yet often unachievable—information need is to retrieve texts based on a description of the content of the text. For example, in the domain of medical research, a user might want to find sentences discussing the efficacy of a specific drug in treating a particular condition, such as “the effectiveness of drug X in managing hypertension.” Or they can go more abstract, and look for “substance abuse in animals” or “a transfer of a disease between two species”. Outside of the hard sciences,

one may want to search the corpus for sentences related to a historical event, such as “an important battle fought during World War II” or “a significant scientific discovery in the field of physics”. An international relation researcher may want to scour a corpus for “one country threatening the economic of another country”, a trader may search for “a transaction involving precious metals”, and a pop-culture journalist may search twitter for “a fight between two celebrities”.

In all these cases, the user is not interested in a definition or a single answer, but for sentences whose content is a specific instantiation of their query (for example, “The studies have shown that a subpopulation of primates chronically consume intoxicating amounts of alcohol” for the “substance abuse in animals” query).

Such retrieval cannot be easily achieved through keyword based retrieval, because the retrieved text is more specific than the description, causing very low lexical overlap. It is also not easily achievable by current “dense retrieval” systems that rely on vector similarity: generic sentence similarity methods (Reimers and Gurevych, 2019; Gao et al., 2021) tend to retrieve texts that are similar to the description, rather than instantiations of it (e.g., a query like “an architect designing a building” should return a sentence like “The original building was designed by architect, William T. Proudfoot” and not “The architect participates in developing the requirements the client wants in the building.”, although the latter is more similar under conventional sentence similarity models). Similarly, systems that are trained to retrieve passages that contain answers to questions (trained, for example, on SQuAD (Rajpurkar et al., 2016, 2018)), beyond being focused on questions rather than assertions, are also focused on specifics rather than abstract situations (questions are of the form “when did Columbus discover America” and not “a discovery on a new land by an explorer”). Models trained on large data

Query: an architect designing a building	
Ours	<ul style="list-style-type: none"> The architect Edwin Forrest Durang designed the building. Léon Eugène Arnal, the chief designer for the architects Magney & Tusler, designed the building. James de Beaujeu Domville helped to design the building. Local architect Robin Chandler designed the building.
Existing	<ul style="list-style-type: none"> The architect of the building is unknown. The building was designed by architect The architect or builder is not known. This article is for the architect.
Query: a company which is a part of another company	
Ours	<ul style="list-style-type: none"> CK Life Sciences International (Holdings) Inc., or CK Life Sciences, is a subsidiary of CK Hutchison Holdings. Pecten (company), a subsidiary of Sinopec WHIRC – a subsidiary company of Wright-Hennepin EM Microelectronic-Marin (subsidiary of The Swatch Group)
Existing	<ul style="list-style-type: none"> Holding company, a company that owns stock in other companies A parent company is a company that owns 51 % or more voting stock in another firm (or subsidiary) When an existing company establishes a new company and keeps majority shares with itself, and invites other companies to buy minority shares, it is called a parent company. Holding (an organisation who owns control of a small group of other companies)
Query: a book that influenced the development of a genre	
Ours	<ul style="list-style-type: none"> The book is now credited with inventing the genre today known as the Regency Historical. It has been cited as an influential classic in the steampunk and dieselpunk genres. It has since become his best-known work, and is considered important in the development of 20th century science fiction in that it is a precursor and likely inspiration to Edgar Rice Burroughs's classic A Princess of Mars (1917), which spawned the planetary romance and sword and planet genres. This book is considered influential in its genre in Russian.
Existing	<ul style="list-style-type: none"> A number of important texts were published in the following decades, developing the genre: The work presents the history of the genre through a discussion of the lives and works of its most important early writers. The latter were significantly important to the development of that genre. Subgenre Book: Like a Genre Book, but focusing on a narrower segment of the full genre.
Query: a change of career path	
Ours	<ul style="list-style-type: none"> He then moved to a manager career. Afterwards, he became an agent full-time. He practice for one year before moving to Streater to start a real estate business. He briefly took on a job as an actuary before embarking into poker.
Existing	<ul style="list-style-type: none"> Otherwise, a change of profession was necessary. As life-expectancy increases, as retirement benefits decrease, and as educational opportunities expand — workers may increasingly find themselves forced to fulfill the goals of one career and then adopt another. It is here in this Midlife Transition that we often find there is an ending of early adulthood as well as individuals making changes in their lives, with the biggest change being the career they are in. Ultimately, you have to take a different direction in your life, in your career.

Figure 1: Top retrieval results from 10 million Wikipedia sentences. **Ours**: the model developed in this work. **Existing**: all-mpnet-base-v2, a strong sentence-similarity encoder.

from search query logs may be more diverse, but are generally not available outside of a few large technology companies.

We show that such retrieval based on description is achievable: given training data consisting of <description, text> pairs, we can train a descriptions encoder and a text encoder that learns to represent items such that the descriptions and the texts they describe are close in embedding space (§4). These vector encodings can then be used in a standard similarity-based retrieval setting.

Figure 1 shows three queries that did not appear in the training data, and their top-4 retrieved items, over a corpus of 10M wikipedia sentences.

To obtain the training data (§3), we observe that the reverse direction of the process, going from a text to its description, is a task that can quite easily be performed either by crowd-workers, or, as we do in this work, by large language models such as GPT-3 (Brown et al., 2020) and Codex (Chen et al., 2021). We thus use the davinci-text-03 model to generate descriptions of sentences that we sampled from Wikipedia, and use the result as our training corpus, which we also make publicly available. Each sentence can accommodate many different descriptions, pertaining to different aspects of the text. We therefore produce five different descriptions for each text, in addition to incorrect descriptions, to be used as negative examples.

This work demonstrates how we can leverage the strengths of LLMs to achieve a task that is not achievable by LLMs alone: retrieving texts, based on descriptions, from large text collections. The description-based retrieval capability we demonstrate in this work can serve as a useful component to enhance discovery ability in many data-intensive domains, and especially in professional domains such legal, medical or scientific search.

2 Description-based Retrieval

The task we define is retrieving sentences that align with a user’s description or specification. This approach is applicable across various domains and can assist users in finding relevant information or examples that match their specific criteria. We compare the description-based similarity with popular existing similarity-based retrieval methods, as well as the related NLP task of recognizing textual entailment (Dagan et al., 2005; Bowman et al., 2015).

Vs. Keyword-based Retrieval Keyword-based retrieval methods rely on exact lexical matches, which makes them inherently weak for retrieval based on abstract descriptions. These methods require users to construct queries using specific keywords, resulting in a laborious and potentially sub-optimal process. For example, to retrieve sentences related to "animals," a user would need to come up with an exhaustive list of animal names, which can be impractical and may lead to incomplete results. Consequently, keyword-based retrieval is ill-suited for retrieving sentences based on abstract descriptions.

Vs. Dense Similarity Retrieval This family of methods, exemplified by SBERT (Reimers and

Gurevych, 2019) encodes sentences based on an objective that encourages sentences with “similar meaning” to have high cosine similarity. Similar meaning, here, is determined by corpora such as Reddit comments (Henderson et al., 2019), SentEval (Conneau and Kiela, 2018) and SNLI (Bowman et al., 2015). However, a description does not have a “similar meaning” to the text it describes, but rather to other descriptions of the same text.

Vs. QA-trained Dense Retrieval These systems are trained to retrieve paragraphs based on a question, in an open-QA setting (Karpukhin et al., 2020). The retrieved paragraphs are then run through a reader component, which attempts to extract the answer from each retrieved paragraph. The training objective is to encode paragraphs to be similar to the questions to which they contain an answer. Question could be seen as similar to descriptions (e.g. “early albums of metal bands” can be served by retrieving for “which metal bands released an early album”), but they also differ in that: (a) it is often cumbersome for a user to rephrase the information need as a question—in the above example, the move to question form is not trivial; (b) questions are often focused on a single entity that is the answer to the question, rather than on a situation involving a relation or interaction between several entities; (c) the kinds of questions in current QA training sets tend to ask about specific, rather than abstract, cases, e.g. asking “which metal band released album Painkiller?” or “what is the first album by Metallica?”.

Vs. Query-trained Dense Retrieval These systems are trained on a collection of <query, document> pairs, which are typically obtained from search engine logs. In the context of academic research, the focus is on the MSMARCO dataset (Bajaj et al., 2016), which contains natural language questions extracted from query logs. However, query logs include many different query types beyond questions, and modern search systems have been reported to incorporate such embedding based results for general queries.¹ In a sense, these subsume the description-retrieval task, but are (a) focused on documents and not on sentences; (b) not focused on this task, so may retrieve also results which are not descriptions; and, most importantly (c)

are mostly based on proprietary data that is only available within a handful of large companies.

Vs. Entailment / NLI <description, text> pairs adhere to the entailment relation between positive <hypothesis, text> pairs in the Textual Inference task (Dagan et al., 2005; Bowman et al., 2015), which is a superset of the <description, text> relation. In theory, NLI based similarity models could perform well on this task. However, in practice they do not perform well, possibly due to the composition of existing NLI datasets. Additionally, they do not usually encode the hypothesis and the premise independently, making efficient indexing difficult.

3 Obtaining Training Data

We use GPT-3 (text-davinci-003) to generate positive and misleading descriptions for sentences from the English Wikipedia dataset.² For each sentence, we generate 5 valid descriptions and 5 misleading descriptions. In total, we generate descriptions for 165,960 Wikipedia sentences. See the Appendix for the exact prompts we use.

Generating more abstract descriptions. While the descriptions we generate do tend to be abstract, to augment the dataset with descriptions of higher abstraction, we randomly select a subset of instances, re-prompt GPT3 with three of the valid descriptions it generated, and ask it to generate abstract versions of them (this prompt is an in-context learning one, the exact prompt appears in the appendix). This results in 69,891 additional descriptions for 23,297 sentences (14.3% of the data). To illustrate the effect of this iterative generation, for the sentence “Civil war resumed, this time between revolutionary armies that had fought in a united cause to oust Huerta in 1913–14.”, one of the original descriptions generated was “A conflict between opposing groups arising from the overthrowing of a political leader”, while the iterative query resulted in the more abstract description “Conflict arose between two sides that had previously been allied.”.

Final dataset. Table 1 shows several examples of the generated data, including the original sentence and pairs of valid and misleading descriptions. The generated data includes a wide range of both positive and misleading descriptions that align with the original sentence and the abstract description. The

¹See, e.g., a report by Google of using BERT <https://blog.google/products/search/search-language-understanding-bert/>

²<https://huggingface.co/datasets/wikipedia>

Sentence	Good Descriptions	Bad Descriptions
Intercepted by Union gunboats, over 300 of his men succeeded in crossing.	A large group of people overcoming a challenge.	A group of people being intercepted while crossing a desert.
Dopamine constitutes about 80% of the catecholamine content in the brain.	A neurotransmitter found in the brain in high concentrations.	A neurotransmitter found in the stomach in high concentrations.
In December 2021, Kammeraad was named in Philippines 23-man squad for the 2020 AFF Championship held in Singapore.	A sportsperson’s inclusion in a squad for a championship.	A soccer player selected for a tournament in the Philippines in 2021.
Around this time, MTV introduced a static and single color digital on-screen graphic to be shown during all of its programming.	A visual element was implemented to enhance the viewing experience.	MTV’s use of a dynamic graphic.
At the signing, he is quoted as having replied to a comment by John Hancock that they must all hang together: “Yes, we must, indeed, all hang together, or most assuredly we shall all hang separately”.	A historical event where a significant figure made a comment about unity.	A joke about the consequences of not working together.
It was said that Democritus’s father was from a noble family and so wealthy that he received Xerxes on his march through Abdera.	A description of a wealthy family’s involvement in a significant event.	A description of a famous leader’s family background.
Heseltine favoured privatisation of state owned industries, a novel idea in 1979 as the Conservatives were initially only proposing to denationalise the industries nationalised by Labour in the 1970s	A political party’s plan to reverse a previous government’s policy.	The effects of privatisation on the economy.

Table 1: Examples of generated data training data, including the original sentence, the good and bad descriptions

positive descriptions accurately capture the main meaning and key concepts of the sentence, while the misleading descriptions contain inaccuracies or irrelevant information. We have randomly divided the data into 158,000 train, 5000 development and 2960 test instances, each composed of a sentence, 5 invalid descriptions and 5-8 valid descriptions. We found the quality of the generated descriptions adequate for training, and for measuring progress during iterative development, which we also confirmed through a human evaluation. We showed 229 descriptions and corresponding sentences to Turkers, asking them to rate on a scale of 4, how well the sentence fits the description. On average the instances were highly rated with a score of 3.69/4, which lies between *The sentence covers most of the main points mentioned in the description* and *The sentence covers everything mentioned in the description*.

However, some of the descriptions do not adequately capture our intended spirit of abstract descriptions of sentences that reflect an information need. Thus, for the purpose of human-evaluation of

quality (Section 5), we manually curate a subset of 200 sentence descriptions from the test set, which we manually verified to reflect a clear information need that would make sense to a human. These were collected by consulting only the descriptions, without the associated sentences they were derived from, or any model prediction based on them.

4 Encoder Training

In order to train our model for the task of aligning sentences with their descriptions, we utilize a pre-trained sentence embedding model and fine-tune it with contrastive learning. During the training process, we represent each sentence and its corresponding valid descriptions using two distinct instances of the model: one as a sentence encoder and the other as a description encoder.

Let S represent a set of sentences, P_s represent the set of valid descriptions associated with a sentence s , and N_s represent the set of negative descriptions for that same sentence s . We encode each sentence and description via the masked language model, resulting in a vector representation

for each token. We use mean pooling over the token vectors of each of the sentence and description pairs to obtain vector representations in \mathbb{R}^d . Specifically, we denote the vector representation of a sentence s as \mathbf{v}_s , the vector representation of a valid description of it as \mathbf{v}_p , and the vector representation of a negative description as \mathbf{v}_n .

To train the encoder, we combine two loss functions: the triplet loss (Chechik et al., 2010) and the InfoNCE loss (van den Oord et al., 2018).

The triplet loss, denoted as $\mathcal{L}_{\text{triplet}}(s)$, is calculated for each sentence s as follows:

$$\sum_{(p,n) \sim P_s \times N_s} \max(0, m + \|\mathbf{v}_s - \mathbf{v}_p\|^2 - \|\mathbf{v}_s - \mathbf{v}_n\|^2) \quad (1)$$

Here, m represents the margin parameter that defines the minimum distance between the positive and negative descriptions. We take $m = 1$. This loss encourages the representation of each sentence to be closer to its valid descriptions than to its invalid descriptions.

The InfoNCE loss, denoted as $\mathcal{L}_{\text{InfoNCE}}(s)$, is computed using a random collection of in-batch negatives (i.e., valid descriptions of *other* sentences in the batch, as well as additional invalid descriptions of other sentences). Let N'_s represent the set of all in-batch negatives sampled from the valid and invalid descriptions of other sentences within the batch. The InfoNCE loss is given by:

$$-\log \left(\frac{\exp(\frac{\mathbf{v}_s \cdot \mathbf{v}_p}{\tau})}{\exp(\frac{\mathbf{v}_s \cdot \mathbf{v}_p}{\tau}) + \sum_{n' \in N'_s} \exp(\frac{\mathbf{v}_s \cdot \mathbf{v}_{n'}}{\tau})} \right) \quad (2)$$

Where \cdot is cosine similarity and τ is the temperature (we take $\tau = 0.1$).

The final loss used for training is a combination of the triplet loss and a scaled version of the InfoNCE loss:

$$\text{Loss}(s) = \mathcal{L}_{\text{triplet}}(s) + \alpha \mathcal{L}_{\text{InfoNCE}}(s) \quad (3)$$

We take $\alpha = 0.1$. Gradients of the embedding vectors of the sentences and their descriptions propagate to the respective encoders.

By optimizing this objective over all sentences in the training set, our encoders learn to effectively align sentences with their respective descriptions in the embedding space. We found that using a combination of the two losses works better than using a single loss alone. We train for 30 epochs with a batch size of 128 and optimize using Adam (Kingma and Ba, 2015).

5 Evaluation

Setting. We encode the training set sentences together with an additional 10 millions Wikipedia sentences using the trained sentence encoder (“the index”). Given a query q , we represent it with the query encoder and perform exact nearest-neighbor search under cosine distance.

Baselines. We evaluate our model against the 3 strongest sentence encoders models³ in the Sentence-Transformer framework (Reimers and Gurevych, 2020),⁴ all-mpnet-base-v2, multi-qa-mpnet-base-dot-v1 and all-distilroberta-v1. All 3 models were finetuned by their creators on diverse sentence-similarity datasets. According to the huggingface website, these models were downloaded millions of times.

Our own model is a fine-tuned version of the pretrained MPnet model (Song et al., 2020) (i.e., we do not use all-mpnet-base-v2, which was further finetuned on similarity datasets, as it yielded worse results in preliminary experiments).

5.1 Qualitative evaluation.

Fig. 1 shows the top results of four queries, alongside the top results based on the state-of-the-art sentence-similarity model all-mpnet-base-v2 for comparison. We release a live demo.⁵ The results clearly show that the retrieved sentences adhere to the requested description. These descriptions were not part of the training set.

5.2 Quantitative evaluation: human judgment of quality

We perform a human evaluation of the sentences retrieved with our method, which we call *abstractsim*, and of the sentences retrieved with the state-of-the-art semantic sentence encoding models. We chose a random set of 200 descriptions from the test set, which we manually verified to be reasonable description-queries a person may be interested in. We then performed crowd-sourced evaluation of retrieval based on these descriptions, comparing our abstract-similarity model to each of the baseline

³According to the evaluations table in https://www.sbert.net/docs/pretrained_models.html, retrieved April 2023.

⁴<https://huggingface.co/sentence-transformers>

⁵<https://github.com/shauli-ravfogel/AbstractSim>

models. While our motivating use-case is high-recall retrieval returning all relevant sentences, this is very hard to evaluate for (and we do not think any current model is close to achieving this goal). Thus, for the purpose of human evaluation we focused on the relevance of the top results.⁶

The evaluation setup is structured as follows.⁷ Crowdworkers are shown a query and 10 sentences, 5 of which are the top-5 retrieved sentences from `abstract-sim` and 5 of which are the top-5 retrieved sentences from one of the baseline (each experiment with another baseline). The 10 sentences are randomly shuffled, and crowdworkers are then asked to select all sentences that they deem a reasonable fit for the query. Each task is shown to three distinct annotators. We aimed at paying crowdworkers \$15 per hour on average. Each query instance is shown to 3 annotators.

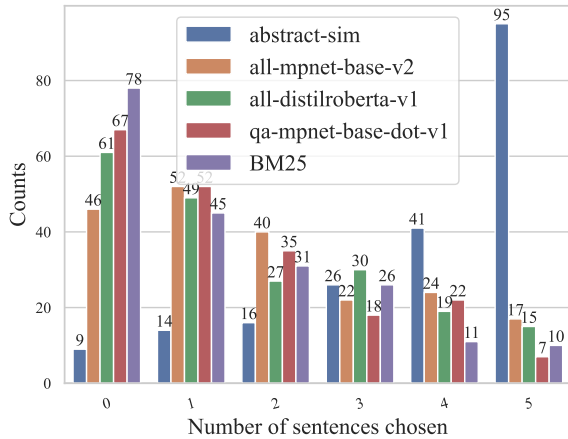


Figure 2: Number of times a given number of sentences was chosen per query instance: Our model (`abstract-sim`), averaged over all 4 baseline evaluations, vs. the baselines.

Metrics We consider two metrics: the average number of results from each model that were selected as relevant, and the number of times a specific number of sentences from a given model was chosen.

Results For evaluation we only count sentences to have been selected as relevant, if they were chosen by at least 2 out of 3 annotators. In Table 2 we show the average number of valid retrieved sentences per each method. The annotators

⁶In our own experiments, we do often observe relevant results also after hundreds of returned item or more.

⁷Screenshots of the annotation interface can be found in the Appendix.

Model	# of selected sents
<code>abstract-sim</code>	3.78 / 5
<code>all-mpnet-base-v2</code>	1.89 / 5
<code>all-distilroberta-v1</code>	1.71 / 5
<code>qa-mpnet-base-dot-v1</code>	1.49 / 5
BM25	1.39 / 5

Table 2: Average number of sentences that crowdworkers deemed to be fitting the query, from a set of 5 retrieved sentences: Our model (`abstract-sim`) vs. the four baselines.

have chosen significantly more sentences from our `abstract-sim` model compared to all 3 baselines, with our model having close to 4 out of 5 sentences deemed as fitting the query on average and the baseline models between 1.39-1.89 sentences. Fig. 2 shows the number of times a given number of sentences was chosen from a given model (where the maximum is 5, that is, all the 5 results for the model were chosen). Notably, in 95/200 of the test cases, 5 sentences were chosen from `abstract-sim`’s results; from the baselines all 5 sentence were only chosen between 7-17 times. Conversely, the baselines show a large number of cases where only 0,1 or 2 sentences were chosen, while these cases are much rarer among `abstract-sim` results.

6 Resulting Models

The training data and similarity models are available on the Huggingface platform (Wolf et al., 2019).⁸ Additionally, we make an online demo for searching sentences based on abstract descriptions available.⁹

7 Conclusions

In this work, we introduce the task of sentence retrieval based on abstract descriptions. We show that current sentence-embedding methods are not a good fit for the task, and provide an alternative model that is significantly better, and which we believe could be of use to many potential users of abstract sentence search. However, we see our model not as a final step, but a first one in a journey towards flexible semantic retrieval.

Searching based on abstract descriptions is made possible due to recent progress in latent semantic representation learning, both pre-trained text en-

⁸The dataset, the sentence encoder and the query encoder.

⁹<https://github.com/shauli-ravfogel/AbstractSim>

coders trained in a self-supervised fashion, and instruct-tuned large language models. We envision a large range of semantic search variants which we hope will be explored in the coming years.

To train our model, we leverage a large language model (specifically, GPT3) to create a training dataset of both accurate and misleading Wikipedia sentence descriptions, enabling us to train a contrastive dual-encoder sentence embedder. Our embedder surpasses robust baselines in a human evaluation trial, underscoring the LLM’s potential for generating tailored training datasets, despite its limitations in direct retrieval tasks. Further research should explore LLMs’ ability to produce diverse abstract descriptions beyond the sentence level, enabling information seeking in large document databases.

Limitations

Our training data, models and experiment are all strictly English-based. More importantly, we observed the following limitation of the resulting similarity model. While it clearly is better than all existing models we compared against at identifying sentences given an abstract description, we also observed the opposite tendency: for some queries, it is not faithful to the provided description. For example, searching for the query “The debut novel of a french author” returns results such as “Eugénie Grandet is a novel first published in 1833 by French author Honoré de Balzac” or “Lanzarote (novel), a novel by Michel Houellebecq”, either mentioning the first time the novel was published, instead of returning mentions of a first novel published by an author; or mentioning novels written by French authors, regardless of whether or not they are their debut novels.

Additionally, this work focuses on sentence level retrieval, while some cases may warrant longer units.

Ethics Statement

As all language technology, the models and data are inherently dual use—they can be used both for good (e.g., to advance human knowledge) or for bad (e.g., for surveillance that is aimed at depression of minority communities). We hope that the benefits outweighs the risks in our case.

According to the terms-of-service of the GPT API, the API output (the collected data and the models we created based on it) should not be used

to compete with OpenAI. We declare we have no such intentions, and ask the users of the data and models to also refrain from doing so.

References

- Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Bhaskar Mitra, Andrew McNamara, Mir Rosenberg, Tri Nguyen, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2016. MS MARCO: A Human Generated Machine Reading Comprehension Dataset. In *InCoCo@NIPS*.
- Samuel 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.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Gal Chechik, Varun Sharma, Uri Shalit, and Samy Bengio. 2010. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11(3).
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.
- Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Proceedings of the First international conference on Machine Learning Challenges: evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, pages 177–190.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.
- Matthew Henderson, Paweł Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulic, et al. 2019. A repository of conversational datasets. *ACL 2019*, page 1.

- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. [Dense passage retrieval for open-domain question answering](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.
- Diederik P. Kingma and Jimmy Ba. 2015. [Adam: A method for stochastic optimization](#). In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don’t know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.
- 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.
- Nils Reimers and Iryna Gurevych. 2019. Sentencebert: 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.
- Nils Reimers and Iryna Gurevych. 2020. [Making monolingual sentence embeddings multilingual using knowledge distillation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33:16857–16867.
- Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. [Representation learning with contrastive predictive coding](#). *CoRR*, abs/1807.03748.
- 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*.

A Appendix

A.1 Prompting

These are the prompts we used to generate the sentence descriptions dataset. The “main prompt” was used to generate 5 valid descriptions and 5 invalid descriptions per sentence. For approximately 14% of the sentences, we re-feed GPT with one of its valid generations and use the “Make-more-abstracts prompt” to generate 3 additional more abstract version of the descriptions.

Main prompt:

Let's write abstract descriptions of sentences. Example:

Sentence: Pilate 's role in the events leading to the crucifixion lent themselves to melodrama , even tragedy , and Pilate often has a role in medieval mystery plays .

Description: A description of a historical religious figure's involvement in a significant event and its later portrayal in art.

Note: Descriptions can differ in the level of abstraction, granularity and the part of the sentence they focus on. Some descriptions need to be abstract, while others should be concrete and detailed.

For the following sentence, write up 5 good and stand-alone, independent descriptions and 5 bad descriptions (which may be related, but are clearly wrong). Output a json file with keys 'good', 'bad'.

Sentence: {sentence}

Start your answer with a curly bracket.

A.1.1 Make-more-abstract Prompt

Sentence: in spite of excellent pediatric health care , several educational problems could be noted in this tertiary pediatric center .

Description: Despite having advanced healthcare resources, certain deficiencies in education were identified at a medical center that serves children.

A very abstract description: The provision of care at a specialized medical center was not optimal in one particular area, despite the presence of advanced resources.

Sentence: {sentence}

Description: {description}

A very abstract description:

A.2 Human-evaluation Interface

This is the interface used for MTurk evaluation:

Instructions (click to expand/collapse)

Thanks for participating in this HIT! Please read the instructions carefully.

In this HIT, you will be shown a **Description** and 10 **Sentences**. The **Description** details what type of sentence we are looking for: Imagine the description is something like a search query for a search machine. The **Sentences** is the result we obtained after searching for sentences that fit the description.

Your task is to **choose** all **Sentences** which you consider good matches for the Search Query/**Description**. Note that the **Sentence** is allowed to contain additional information, not mentioned in the **Description**, as long as it covers what has been requested in the **Description**.

Please take care to not submit responses that are uninformed by the instructions.

Description:

#{description1}

1. Choose all **retrieved sentences** that fit the **description**

- ☐ #{sentence1}
- ☐ #{sentence2}
- ☐ #{sentence3}
- ☐ #{sentence4}
- ☐ #{sentence5}
- ☐ #{sentence6}
- ☐ #{sentence7}
- ☐ #{sentence8}
- ☐ #{sentence9}
- ☐ #{sentence10}

(Optional) Please let us know if anything was unclear, if you experienced any issues, or if you have any other feedback for us.

This is the interface with an instantiated descriptions and 10 retrieved sentences (5 from baselines and 5 from our model, presented in random order).

Description:

A period of difficulty and sorrow for an individual.

1. Choose all **retrieved sentences** that fit the **description**

- ☐ For some years in the period 1945–1974 there was an individual college championship.
- ☐ It was a period of great political difficulty in Italy.
- ☐ His personal life at this time was filled with tragedy.
- ☐ An tak it not in sorrow;
- ☐ Gauss plunged into a depression from which he never fully recovered.
- ☐ Attwater suffered for several days afterwards, though.
- ☐ The time of suffering and illness
- ☐ During this time, however, Charlesfort had fallen into despair.
- ☐ An wed your sons wi sorrow;
- ☐ The difficulty of properly assessing the value of an individual gem-quality diamond complicates the situation.

This is the interface we used for assessing the coverage of the GPT3 generated description and its corresponding sentence.

Description:**`\${description1}`****Retrieved
Sentence:****`\${sentence1}`**

1. Coverage: How well does the **retrieved sentence** fit the description? (Note: The descriptions should capture some aspect of the sentence, but they don't need to fully describe all the facets of the sentence: i.e. the sentence is allowed to contain additional information not mentioned in the description and should not be penalized for it.)

- ☐ **1.** The retrieved sentence is **not relevant at all** with respect to the description.
- ☐ **2.** The retrieved sentence contains **minor** elements mentioned in the description.
- ☐ **3.** The retrieved sentence covers **some of the points** mentioned in the description.
- ☐ **4.** The retrieved sentence covers **most of the main points** mentioned in the description.
- ☐ **5.** The retrieved sentence covers **everything** mentioned in the description.