Project Literature Review, Solution Proposal
JaMiMaKa, topic: Analysis of Questions, Autumn 2023

Kacper Grzymkowski	Jakub Fołtyn	Marceli Korbin	Mikołaj Malec	Dr. Anna Wróblewska
MSc student	MSc student	MSc student	MSc student	supervisor
WUT	WUT	WUT	WUT	lecturer at WUT
kacper.grzymkowski .stud@pw.edu.pl	01151388 @pw.edu.pl	01142124 @pw.edu.pl	01142129 @pw.edu.pl	anna.wroblewska1 @pw.edu.pl

Abstract

This document represents the preliminary stage of work for the project regarding the Analysis of Questions topic. In it, we the authors present literature review for papers related to the project's topic, as well as some datasets reviews and proposals for further research as well as solutions for this very project. It is worth noting, however, that due to the project's topic complexity, as well as delays in obtaining the appropriate dataset for the project, the solution proposals introduced in this document may be subject to change.

1 Credits

This project is being realised in cooperation with prof. Yoed Kenett from Israel Institute of Technology in Haifa, Israel. The datasets used in this project have been made available by prof. Siew Ann, NTU as well as Sebastijan Macek from STA.

2 Introduction

People tend to ask a lot of questions. Some of them may be simple and be utilized to extract some very basic knowledge, such as ”What's the weather like today?”, or ”Can you help me find the post office?”. On the other hand, we have questions that tackle much more complicated topics, answers to which may need some additional information tackling philosophy or other sciences, such as ”What is happiness?” or ”How does it work?”. It is those kinds of questions that bring us closer to the second major (although latent) topic of our project: creativity. Scientists such as prof. Yoed Kenett from Israel Institute of Technology believe, that asking questions plays a critical role in the creative process. Formulating questions, Yoed says, helps people characterize and define the problem – which is the first step in creative process. What is more, further question asking and answering may help increase one's creativity potential. It is also these questions, and the process of their formulation, that help scientists better understand and study the very concept of human creativity. In this project, our goal is to develop a method (using State-Of-The-Art Natural Language Processing (Chowdhary and Chowdhary, 2020) solutions) for clustering questions based on their structure and difficulty, as well as to (potentially) develop measures of question complexity. In this document, we will focus on reviewing previous works tackling the topic of questions (in a very broad sense) as well as some possible datasets that will be utilized in our project. Finally, we will propose some possible solutions to the project's topic.

3 Literature review

In this section we will present different approaches and areas of the topic of questions tackled by some of the previous works.

3.1 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a powerful and widely used statistical model for understanding and analyzing collections of discrete data, particularly in the context of text corpora. It is considered a generative probabilistic model, meaning that it helps explain how the data might have been generated. At its core, LDA is a three-level hierarchical Bayesian model that seeks to uncover the latent structure hidden within a collection of documents. This structure revolves around the concepts of topics and the distribution of words within those topics. In LDA, a document is viewed as a mixture of topics. It assumes that a document is not about just one topic, but rather a combination of several topics. These topics are latent, meaning they are not directly observed but are inferred from the text. Each topic, in turn, is viewed as a distribution over words. This means that a topic is characterized by a set of words that are more likely to occur when discussing that particular topic. These word distributions are also hidden variables that LDA aims to uncover. The model assumes that documents are generated in a probabilistic manner. To be more precise, for each document, LDA assigns a distribution of topic probabilities. These probabilities indicate the likelihood of a particular document containing certain topics. For example, a news article about technology might have a high probability of containing topics related to ”technology” and ”innovation,” but it may also have smaller probabilities for other topics like ”politics” or ”health.” Finally, the actual words in the document are generated based on the topics. For each word in the document, LDA selects a topic from the distribution of topics specific to that document and then selects a word from the distribution of words for that topic. In essence, LDA seeks to reverse-engineer the topic structure that might have generated a given collection of documents. It does this through statistical inference, trying to find the most likely set of topics and their associated word distributions that would have created the observed documents. The LDA model can be trained on a large text corpus to discover these latent topics and word distributions, providing valuable insights into the content and themes present in the data. LDA's generative nature allows it to be a versatile tool for various natural language processing tasks, such as document clustering, topic modeling, text summarization, and even recommendation systems. By understanding the underlying topics within a collection of documents, LDA enables researchers, data scientists, and analysts to uncover patterns, extract meaningful information, and gain a deeper understanding of the content within textual data. In summary, Latent Dirichlet Allocation is a powerful probabilistic model that offers a structured approach to understanding the hidden thematic structure within text corpora. Its ability to reveal latent topics and their associated word distributions has made it an indispensable tool for various applications in the field of natural language processing and text analysis.

3.2 BERT

BERT, or Bidirectional Encoder Representations from Transformers, is a breakthrough in natural language processing and understanding that has revolutionized the way machines comprehend and generate human language. BERT is not a generative model like LDA but rather a pre-trained transformer-based model that excels at various language understanding tasks. Unlike earlier models that processed text in a unidirectional manner, BERT introduced bidirectional context. It understands words in relation to their entire context within a sentence. In other words, BERT considers both the words that come before and after a given word, allowing it to capture the full meaning and nuances of the language. This bidirectionality is crucial in understanding the context, making it highly effective in tasks like sentiment analysis, question answering, and language translation. BERT is built upon the transformer architecture, which has proven to be highly effective for a wide range of natural language processing tasks. The transformer architecture is designed to handle sequential data, like text, and is based on the concept of attention mechanisms. BERT leverages this architecture to process input data through a stack of self-attention layers, which enables it to model the relationships between words in a sentence efficiently. BERT's power comes from pre-training on massive amounts of text data. During pre-training, it learns to predict missing words in a sentence and understand the context in which each word appears. This pre-trained model is then fine-tuned for specific downstream tasks. Fine-tuning involves training the model on smaller, task-specific datasets, which makes it adaptable to various applications. This fine-tuning process enables BERT to excel in tasks such as text classification, named entity recognition, and machine translation. BERT has been trained in multiple languages, making it a valuable resource for multilingual applications. It can understand and generate text in a wide range of languages, which is crucial for global businesses and organizations that need to process text in different languages. BERT's deep bidirectional learning enables it to capture semantic relationships between words and phrases. This means it can understand not only the surface meaning of words but also their contextual significance. It's capable of recognizing synonyms, antonyms, and even nuances in sentiment, which is especially important in tasks like sentiment analysis and language generation. BERT introduced the concept of transfer learning to NLP, which allows models to leverage knowledge learned from one task to perform better on other, related tasks. This has greatly reduced the amount of labeled training data needed for many NLP applications and accelerated progress in the field. BERT has become a cornerstone of modern natural language processing, and its availability as a pre-trained model has lowered the barriers to entry for developing NLP applications. Researchers, developers, and organizations can take advantage of BERT's pre-trained representations and fine-tuning capabilities to quickly build and deploy powerful language understanding systems. In summary, BERT represents a major advancement in natural language processing by bringing bidirectional context, transfer learning, and the transformer architecture together. Its versatility and ability to understand the intricate details of language have made it an indispensable tool for a wide range of NLP tasks, from sentiment analysis to machine translation and beyond.

3.3 Seed-Guided algorithm

The Seed-Guided algorithm, often referred to as "SeedTopicMine," is a cutting-edge approach in the realm of topic modeling and text analysis. This innovative technique harnesses the power of seed words, which are manually provided keywords or terms, to guide the discovery and exploration of topics within a collection of documents. SeedTopicMine represents a fusion of both supervised and unsupervised learning, offering a more directed and controlled way to identify specific topics of interest. At the heart of SeedTopicMine are the seed words. These are carefully selected terms or keywords that serve as the initial cues for the algorithm. Researchers or domain experts typically provide these seed words, which represent the topics they are interested in exploring. The algorithm will use these seeds as a starting point to uncover and expand upon related topics within the document collection. SeedTopicMine combines aspects of both supervised and unsupervised learning. While traditional unsupervised topic modeling methods like Latent Dirichlet Allocation (LDA) discover topics without any predefined guidance, SeedTopicMine leverages the seed words to kickstart the process. This approach ensures that the algorithm focuses on the specific topics of interest, making it highly suitable for applications where prior knowledge of the subject matter is available. SeedTopicMine begins by associating the seed words with the documents in the collection. It then uses these associations to explore and discover related terms and phrases within the text. By iteratively expanding upon the initial seed words, the algorithm identifies a broader set of terms that define the topics under investigation. This process allows for a more granular and focused understanding of the topics present in the corpus. The algorithm is adaptive and iterative, meaning that it learns from the discovered terms and relationships. As it identifies new terms that are related to the seed words, it refines its understanding of the topics and can incorporate additional terms into the topic models. This adaptability enables the algorithm to handle evolving or dynamic topics within the document collection. SeedTopicMine is especially valuable in domain-specific applications where the language and terminology used are specialized. By providing seed words that are specific to the domain, experts can guide the algorithm to extract topics that are relevant and meaningful within that context. SeedTopicMine has a wide range of applications, including content recommendation, information retrieval, content summarization, and domain-specific knowledge extraction. It is particularly useful in scenarios where a targeted and precise understanding of the content is required, such as in legal documents, healthcare records, and academic literature. In summary, the Seed-Guided algorithm "SeedTopicMine" offers a novel and powerful approach to topic modeling and text analysis. By leveraging seed words to initiate and direct the topic discovery process, it provides a more controlled and focused way to explore topics within a document collection. This adaptability and domain specificity make SeedTopicMine an invaluable tool for uncovering meaningful insights and knowledge from textual data in specialized domains and applications.

3.4 Sentence Embedding

Sentence Embedding is a crucial technique in natural language processing that plays a significant role in various text analysis tasks, including text classification, semantic similarity, and information retrieval. It aims to transform a sentence or a piece of text into a fixed-length vector representation, preserving its semantic meaning and context. Several approaches have been developed to create sentence embeddings, and here, we'll explore three prominent methods: Averaging Word Embeddings, Pre-trained Models like BERT, and Neural Network-Based Approaches. Averaging word embeddings is a simple yet effective technique to obtain sentence embeddings. In this method, each word in a sentence is represented as a word embedding vector, typically obtained from pre-trained word embeddings like Word2Vec or GloVe. The sentence embedding is then calculated by averaging the word embeddings of all words in the sentence. While straightforward, this method often captures the overall meaning of the sentence, making it useful for tasks where context may not be as critical. However, it may lose nuances and complex sentence structures. Pre-trained models like BERT (Bidirectional Encoder Representations from Transformers) have revolutionized the field of sentence embedding. BERT, as a contextualized language model, learns to represent words based on their surrounding context. To obtain sentence embeddings using BERT, one can simply feed the entire sentence to the model, and it returns contextualized embeddings for each word. These embeddings are then pooled or aggregated to create a single sentence embedding. BERT embeddings are highly contextual and capture the meaning of the sentence with intricate details, making them suitable for a wide range of tasks, from sentiment analysis to question answering. Neural network-based approaches involve training specific models to generate sentence embeddings. These models can range from simple feedforward neural networks to more complex architectures like Siamese networks. Siamese networks, for example, are trained to compare sentence pairs and produce embeddings that reflect the similarity or dissimilarity between sentences. Neural network-based approaches offer flexibility in designing models tailored to specific tasks and datasets, making them suitable for applications where fine-tuned control is needed over the embedding process. Each of these approaches has its advantages and is applicable in different scenarios. Averaging Word Embeddings method is quick and easy, making it a practical choice for applications where computational resources are limited. It's particularly useful for tasks that require simplicity and speed. Pre-trained Models like BERT: BERT-based embeddings offer state-of-the-art performance in many NLP tasks. They excel at capturing context, nuances, and semantic meaning. These embeddings are highly recommended for applications where understanding the full context of a sentence is crucial. Neural Network-Based Approaches: These approaches provide a middle ground, offering a balance between simplicity and context preservation. They can be fine-tuned for specific tasks and are suitable when a certain level of customization is required. In summary, sentence embedding is a fundamental technique in NLP, and the choice of method depends on the specific requirements of the task. Whether through simple averaging, leveraging pre-trained models like BERT, or utilizing neural network-based approaches, the goal remains the same: to convert text into meaningful numerical representations that can be used for a wide range of natural language processing applications.

3.5 WTC-corpus

When thinking about the topic of questions in NLP, one may usually consider the task of question answering. While this is a perfectly valid task, in our project we would like to go beyond it and focus on processing and acquiring information about the questions themselves. That is why works such as article titled ”What makes us curious? Analysis of a corpus of open domain questions” (Xu et al., 2021) was especially important in our research. There, authors propose a dataset consisting of over 10,000 questions (8,000 after filtering) asked by various residents of Bristol, England. This dataset is also called the **WTC-corpus**. The goal of that article was to study the curiosity and capture the thinking processes of Bristolians. To achieve this, in addition to question answering, authors also considered the tasks of question topic classification and question equivalence/similarity detection. Authors' intention was to create singular model capable of performing all of these task. For this purpose, **BERT** (Devlin et al., 2018) model was utilized, or, more precisely – **S-BERT** (Reimers and Gurevych, 2019), a modification of BERT that captures sentence similarity and provides embedding for a given sentence. This model was then fine tuned utilizing some additional datasets specialized in each of the above mentioned tasks. The resulting (after fine-tuning) model was called **QBERT**.

For question topic classification and question answering (in the form of choosing an appropriate answer from a set of given answers) authors used simple classification with Softmax function applied to a element-wise difference of embeddings values. For the similarity detection task, and question answering in the variant of retrieving possible answers from wikipedia articles summaries, authors used cosine similarity. In general, models created yielded promising results, although their performance depended heavily on the configuration of datasets used for fine-tuning the QBERT model. These results are promising and may give us some preliminary proposals on how to tackle the tasks of question clustering and topic classification. Another interesting and potentially useful action proposed by the authors was dividing questions into two categories: factual and counterfactual questions. The first group consists of so-called ”WH-questions”, meaning questions involving words ”when”, ”who”, ”where”, ”why” etc. while the second group usually includes question with the word ”if”. Questions from the counterfactual group, according to authors, usually require more complex answers and may give a better insight into a person’s reasoning. That is why such a division may also be advantageous for our project.

3.6 Topic modeling

Topic modeling, while not being directly connected to questions, may still prove to be most beneficial for our project. It is an unsupervised technique used to identify natural topics in text (Blei and Lafferty, 2009). It may be especially helpful in the task of question clustering, as intuitively questions regarding similar topics should be clustered together.

One of the most popular methods of topic modeling is **latent Dirichlet allocation** (**LDA**)(Blei et al., 2003). It has been used in another article that we would like to mention in this section: ”Text Mining of Open-Ended Questions in Self-Assessment of University Teachers: An LDA Topic Modeling Approach” (Buenano-Fernandez et al., 2020).

In this paper, as the title suggest, authors apply the LDA topic modeling method to a dataset of open-ended questions (and their answers). That dataset was created from the online surveys for self-assessment of teachers in an Ecuadorian university. The approach for this task may be considered quite straight-forward: authors simply create a term-document matrix from a previously preprocessed database, and then apply LDA technique to retrieve various topics from documents and assign specific terms to them. In this way, it is possible to retrieve the frequency of specific topics for specific documents. The relevance of found topics is then assessed by an expert.

While the presented pipeline may be basic, the main strength of this article are the various visualisations, depicting the possible clusterings based on retrieved and assigned topics. They may serve as possible inspiration for the latter stages of our project.

While LDA is a popular method, authors of (Zhang et al., 2023) argue that it may retrieve semantically general topics that may not align well with users’ specific interests. To counter this, they propose a seed-based approach, that would allow users to extract topics based on some provided seeds. In reality, this approach combines several different approaches (such as **Skip-Gram Word Embeddings** (Mikolov et al., 2013), **Pretrained Language Model Representations** and **Topic-Indicative Documents** to discover a set of more fine-grained topic-indicative terms. These terms may then be used to retrieve topic-indicative sentences inside documents. Authors also argue that their approach combines multiple types of contexts (from all of the above mentioned approaches), which contributes to a more reliable choosing of topic-indicative terms. This process has been described in more detail in one of the previous subsections.

We consider this article to be potentially useful in our project, especially since the approach presented is described in great detail, and, as claimed by the authors, may lead to more reliable results then a more popular LDA-based approach.

3.7 Transformers and LLMs

The transformer architecture, introduced in (Vaswani et al., 2017), facilitated great advances in natural language processing in the past few years. Tasks such as text classification, machine translation, cognitive dialogue systems or information retrieval achieve phenomenal results, however this has come with a heavy computational costs (Singh and Mahmood, 2021). Training these large models requires immense amounts of data, computational resources and technical expertise, which makes them out of reach for most researchers. Indeed, a lot of research has gone into making models more efficient and more available (Singh and Mahmood, 2021). This research however mostly focuses on inference efficiency, which while important for commercial use, is less important for our purpose, as we need to focus on creating a new system, rather than making an existing one more efficient.

Another approach commonly used to deal with the computational needs of the language models is pre-training and fine-tuning, introduced in (Devlin et al., 2018) and (Radford et al., 2018). This approach allows creation of "general" pretrained models which are created using unsupervised methods, which then can be fine-tuned to specific tasks. Importantly, the process of fine tuning is much less data and computationally intensive than creating training "from scratch". This approach is still commonly used, but it has some issues. In recent years, most advancements in language models usually are achieved by creating a bigger model, and this can be easily demonstrated by using the successors to the **GPT** model, in which the first one has 110 million parameters (Radford et al., 2018), while the **GPT-2** has 1.5 billion parameters (Radford et al., 2019), **GPT-3** with 175 billion parameters (Brown et al., 2020), and finally **GPT-4** which is rumored to be around 1.76 trillion parameters (Schreiner, 2023). The sheer size of the models makes even the "lightweight" approach of fine-tuning more and more difficult. While some work is being done to help with fine-tuning such as Low Rank Adaptation (**LoRA**) (Hu et al., 2021), this is not enough. Another aspect to consider is that fine-tuning still requires quite a lot of data, which can be very expensive to acquire, especially if expert knowledge is needed.

These considerations, combined with increasing efficacy of language models is why another approach was devised, the so-called "prompt-engineering", in which no updates to the model weights are applied, and the models is simply "primed" with the needed information by normal interaction with the model (Brown et al., 2020). These are split into zero-shot learning, in which the language model is simply provided with a single prompt, such as "Translate the following sentence into Spanish", one-shot learning, in which the model is additionally provided with an example, such as "Hello → Hola", and few-shot learning, in which the model is provided with more examples than one, but still way lower than the amount needed for fine-tuning, usually less than 10. This opens a way to very quickly and very cheaply create specialized models, and can be theoretically done with no expertise in natural language processing, deep learning or programming.

However, this approach has limitations, in particular with the so-called "hallucinations", where a model is not retrieving factual information, but is fabricating the information as it generates more text, and presents it as fact. One area where this downside is not as prominent is when we need to answer "common-sense" questions, which are often poorly defined and it can be even difficult to assess their veracity. Using our project as an example: is a question "What is the meaning of life?" difficult? The outputs of such a query can be viewed as just an opinion, while it shouldn't be treated as the ground truth, it could be used to help inform a final decision.

A serious limitation to consider is that many of the newer models aren't available publicly, and are only available via a designated channel, usually after a commercial agreement and an agreement to share the data. This can violate non-disclosure agreements, which is less than ideal for our purposes, as we will likely be working on data that is covered by such an agreement. However, open source models and other source/weight available models such as **LLaMA2** (Touvron et al., 2023) or **Mistral7B** (Jiang et al., 2023) can prove to be viable alternatives.

4 Proposed datasets

This section introduces several datasets, which we consider working on during the project.

4.1 R. Tatman's Question-Answer Dataset

This dataset is maintained by Rachael Tatman at Kaggle (Tatman, 2017). It contains three question files, divided by year of students, and 690,000 words of text from Wikipedia, with which the questions were generated. Question files consist of both questions and answers, along with the Wikipedia article of their origin and levels of the question difficulty, assigned by the questioner and the answerer, not necessarily of the same value.

4.2 Question-Answer Jokes From Reddit

The dataset was prepared and uploaded by Jiří Rožnovjak (Rožnovjak, 2017). It provides jokes in the question-answer from, retrieved from the r/Jokes subreddit, a part of the Reddit platform. Each row contains a question-answer pair and an ID; the data is collected from the years 2008–2016.

4.3 Stanford Question Answering Dataset

It is a reading comprehension dataset (sta, 2016) prepared by the Stanford University, which contains around 100,000 questions asked by crowdworkers on over 500 Wikipedia articles, along with their answers. The dataset is divided into two JSON files.

4.4 Large Question Answering Datasets

This is a collection of large question-and-answer datasets (Matthias Hertel, 2020), which additionally contains several question generation systems; all the papers describing them were published in 2010 or later.

4.5 Quora's Question Pairs

Question Pairs was developed in order to deal with duplicate questions on the Quora site. Unlike other datasets mentioned in the section, this dataset groups questions in potential duplicate pairs and assigns them a value indicating whether the pair actually contains duplicates. In the dataset, there are more than 400,000 rows.

4.6 Other potentially obtainable datasets

4.6.1 Yahoo! Answers

The Yahoo! Answers dataset (mentioned in (Zhang et al., 2016)) was based on the 4.5 million question/answer corpus and contains data grouped into 10 categories by topic, functioning as a classification dataset. In order to obtain the data, we should consider using the Yahoo! Webscope program, with which the authors of the cited paper were able to prepare data.

4.6.2 WTC Corpus

Already mentioned and discussed in the Literature review; we are still in the process of negotiating the access to the dataset.

5 Solution concept and proposal

Our project is quite ambitious and we need to consider our very limited time-frame. While we think it would be an interesting option to fine-tune a model such as **BERT** for this specific task, it's unfortunately not really available to us. This is why we would like to mostly use ready made models to data-mine the questions and answers and then pass the results into a classical ML model algorithm, which we feel is a safer option. Examples of data-mining we would perform would include relatively simple operations, such as counting the number of uncommon words, to more complex techniques, such as calculating sentence embeddings for the question and answers, using named entity recognition, topic-modelling or even using prompt-engineered queries with LLMs. This architecture would allow us to more easily adapt different aspects of the questions while working in the team in parallel, which will help with the tight schedule.

Splitting question-only and question&answer into two separate analyses might be an interesting option, however we will focus on the Q&A option, and consider question-only if we have enough time. Another possibility is to measure the general impact of each data-mining methods on the clustering. Many of the techniques we plan on using generate discrete data, which needs to be processed further to obtain data which is "cluster" friendly. While those are interesting considerations, we will likely not have enough time in the first part of the project to fully study.

The data situation is not great. While we have some great datasets, they're completely unlabeled for our purposes. We also can't realistically request large amounts of labels from psychologists, as we do not have the funding for such an endeavour. This realistically leaves us with one main option of performing clustering on the questions, with possible post-hoc analysis to try and deduce the patterns present in the questions, and attempt to label the clusters ourselves with varying levels question complexity. This post-hoc analysis would obviously be limited to our knowledge, therefore we have to consider whether we can fully classify questions in the full scale of Bloom's taxonomy, and whether even a "simple"/"complex" binary classification would be possible to perform by a team that did not have any experience in psychology.

That being said, if it was possible to acquire some amount of labeled data, we would be able to refine the model much more effectively, using partially unsupervised techniques. One possibility of acquiring such labels could be use the dataset itself, for example we can assume that generally questions asked during a lecture would be more complex, than ones asked on an internet forum.

The architecture diagram for our proposal can be seen on figure 1. On the left, we have our data, that will need to be prepared and formatted in a consistent way across the different datasets we decide to use. This data will then be processed as part of multiple separate mining techniques. The techniques shown are primarily for demonstration purposes, and the list is likely to grow and change as work on the project progresses. Afterwards, the results of the models will be forwarded into a clustering algorithm which we will perform post-hoc analysis on. This will likely involve sampling of the clusters to see how they're split, as well as some visualisations of the global trends, for example in which clusters is a topic mentioned.

6 Conclusion

In conclusion, we think the topic is very interesting and can shed more light on the very interesting field of study – the cross between psychology and technology. The AGI (Artificial General Intelligence), a computer system that is equal or superior to a human, is a holy grail of AI. Sometimes it's even treated as something to be afraid of, as for the first time in our species' existence we would have a more intelligent being in close proximity. This problem however is rarely considered from a psychological standpoint, which we think is a shame. If we work to understand why a large language model behaves the way it does, we could try to use it back in psychology where we don't know exactly what goes on inside the brain either.

References

David M Blei and John D Lafferty. 2009. Topic models. In *Text mining*, pages 101–124. Chapman and Hall/CRC.

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

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.

Diego Buenano-Fernandez, Mario Gonzalez, David Gil, and Sergio Luján-Mora. 2020. Text mining of open-ended questions in self-assessment of university teachers: An lda topic modeling approach. *Ieee Access*, 8:35318–35330.

KR1442 Chowdhary and KR Chowdhary. 2020. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Nathalie Prange Matthias Hertel. 2020. Large question answering datasets.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

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.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Jiří Rožnovjak. 2017. Question-answer jokes.

Maximilian Schreiner. 2023. Gpt-4 architecture, datasets, costs and more leaked. Blog post on the-decoder.com.

Sushant Singh and Ausif Mahmood. 2021. The nlp cookbook: Modern recipes for transformer based deep learning architectures. *IEEE Access*, 9:68675–68702.

2016. Stanford question answering dataset.

Rachael Tatman. 2017. Question-answer dataset.

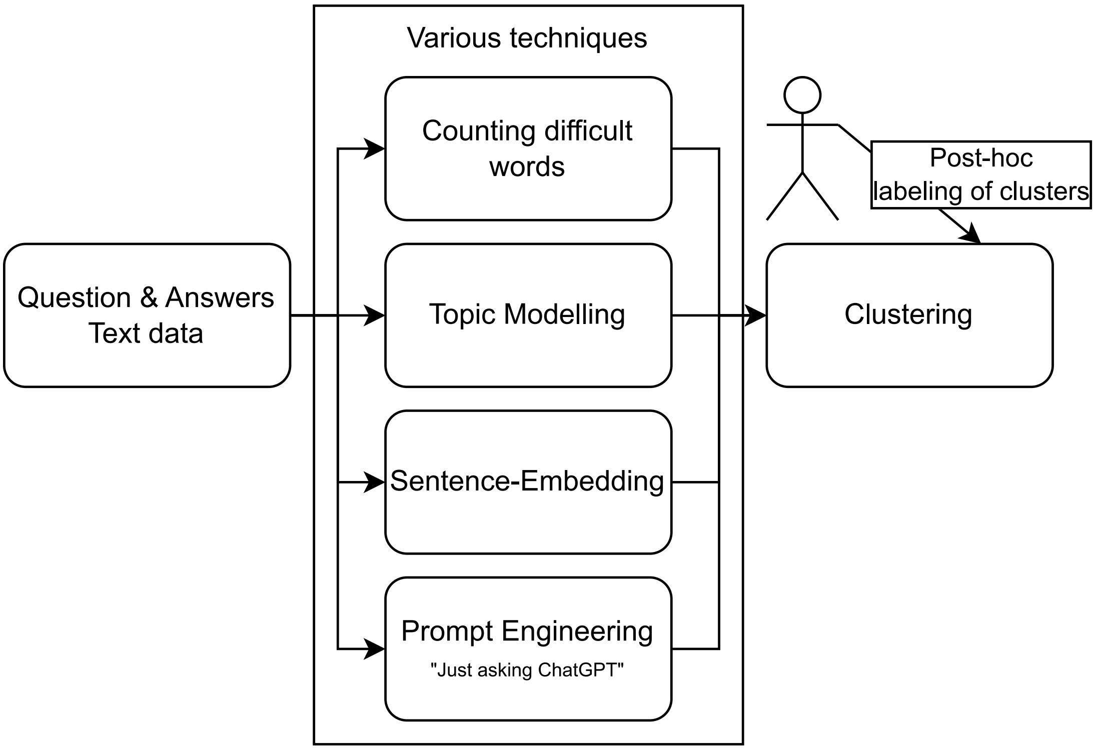

Figure 1: Overview diagram of the project. The techniques shown are primarily for demonstration purposes, as we begin work on the project we will

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Zhaozhen Xu, Amelia Howarth, Nicole Briggs, and Nello Cristianini. 2021. What makes us curious? analysis of a corpus of open-domain questions. *arXiv preprint arXiv:2110.15409*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2016. Character-level convolutional networks for text classification.

Yu Zhang, Yunyi Zhang, Martin Michalski, Yucheng Jiang, Yu Meng, and Jiawei Han. 2023. Effective seed-guided topic discovery by integrating multiple types of contexts. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pages 429–437.