



MLCon
CONFERENCE & TRAINING

Exploring OpenAI's Embeddings

A Guide to Advanced
Natural Language Processing

WHITEPAPER

CONTENT

OpenAI Embeddings

3

OpenAI New Models: The Game-Changer For Natural Language Processing (NLP)

by Rainer Stropek

Language Models as Moral Machines

8

Can ChatGPT influence our value judgments? An interview with Dr. Matthias Uhl

by Matthias Uhl

MathML: Fundamentals and Practice for Formulas

11

Fabulous Formulas

by Dr. Thomas Meinike

Building a Proof of Concept Chatbot with OpenAI's API, PHP and Pinecone

17

Transforming Customer Support with AI

by Daniel Archer

AI is a Human Endeavor

23

An interview on AI regulation, existential threats narratives, and the need for public discourse about AI

by Mhairi Aitken

OpenAI New Models: The Game-Changer For Natural Language Processing (NLP)

OpenAI Embeddings

Embedding vectors (or embeddings) play a central role in the challenges of processing and interpretation of unstructured data such as text, images, or audio files. Embeddings take unstructured data and convert it to structured, no matter how complex, so they can be easily processed by software. OpenAI offers such embeddings, and this article will go over how they work and how they can be used.

by [Rainer Stropek](#)

Data has always played a central role in the development of software solutions. One of the biggest challenges in this area is the processing and interpretation of unstructured data such as text, images, or audio files. This is where embedding vectors (called embeddings for short) come into play – a technology that is becoming increasingly important in the development of software solutions with the integration of AI functions.

Embeddings are essentially a technique for converting unstructured data into a structure that can be easily processed by software. They are used to transform complex data such as words, sentences, or even entire documents into a vector space, with similar elements close to each other. These vector representations allow machines to recognize and exploit nuances and relationships in the data. Which is essential for a variety of applications such as natural language processing (NLP), image recognition, and recommendation systems.

OpenAI, the company behind ChatGPT, offers models for creating embeddings for texts, among other things. At the end of January 2024, OpenAI presented new versions of these embeddings models, which are more powerful and cost-effective than their predecessors. In this article, after a brief introduction to embeddings, we'll take a closer look at the OpenAI embeddings and the recently introduced innovations, discuss how they work, and examine how they can be used in various software development projects.

Embeddings briefly explained

Imagine you're in a room full of people and your task is to group these people based on their personality. To do this, you could start asking questions about different personality traits. For example, you could ask how open someone is to new experiences and rate the answer on a scale from 0 to 1. Each person is then assigned a number that represents their openness.

Next, you could ask about another personality trait, such as the level of sense of duty, and again give a score between 0 and 1. Now each person has two numbers that together form a vector in a two-dimensional space. By asking more questions about different personality traits and rating them in a similar way, you can create a multidimensional vector for each person. In this vector space, people who have similar vectors can then be considered similar in terms of their personality.

In the world of artificial intelligence, we use embeddings to transform unstructured data into an n-dimensional vector space. Similarly how a person's personality traits are represented in the vector space, each point in this vector space represents an element of the original data (such as a word or phrase) in a way that is understandable and processable by computers.

OpenAI Embeddings

OpenAI embeddings extend this basic concept. Instead of using simple features like personality traits, OpenAI models use advanced algorithms and big data to achieve a much deeper and more nuanced representation of the data. The model not only analyzes individual words, but also looks at the context in which those words are used, resulting in more accurate and meaningful vector representations.

Another important difference is that OpenAI embeddings are based on sophisticated machine learning models that can learn from a huge amount of data. This means that they can recognize subtle patterns and relationships in the data that go far beyond what could be achieved by simple scaling and dimensioning, as in the initial analogy. This leads to a significantly improved ability to recognize and exploit similarities and differences in the data.

Individual values are not meaningful

While in the personality trait analogy, each individual value of a vector can be directly related to a specific characteristic – for example openness to new experienc-

$$\text{Cosine Similarity} = \frac{A \cdot B}{\|A\| \times \|B\|}$$

Figure 1 -Cosine similarity

$$\text{Cosine Similarity} = \frac{A \cdot B}{\|A\| \times \|B\|}$$

Figure 2 – Calculation of cosine similarity

es or a sense of duty – this direct relationship no longer exists with OpenAI embeddings. In these embeddings, you cannot simply look at a single value of the vector in isolation and draw conclusions about specific properties of the input data. For example, a specific value in the embedding vector of a sentence cannot be used to directly deduce how friendly or not this sentence is.

The reason for this lies in the way machine learning models, especially those used to create embeddings, encode information. These models work with complex, multi-dimensional representations where the meaning of a single element (such as a word in a sentence) is determined by the interaction of many dimensions in vector space. Each aspect of the original data – be it the tone of a text, the mood of an image, or the intent behind a spoken utterance – is captured by the entire spectrum of the vector rather than by individual values within that vector.

Therefore, when working with OpenAI embeddings, it's important to understand that the interpretation of these vectors is not intuitive or direct. You need algorithms and analysis to draw meaningful conclusions from these high-dimensional and densely coded vectors.

Comparison of vectors with cosine similarity

A central element in dealing with embeddings is measuring the similarity between different vectors. One of the most common methods for this is cosine similarity. This measure is used to determine how similar two vectors are and therefore the data they represent.

To illustrate the concept, let's start with a simple example in two dimensions. Imagine two vectors in a plane, each represented by a point in the coordinate system. The cosine similarity between these two vectors is determined by the cosine of the angle between them. If the vectors point in the same direction, the angle between them is 0 degrees and the cosine of this angle is 1, indicating maximum similarity. If the vectors are orthogonal (i.e. the angle is 90 degrees), the cosine is 0, indicating no similarity. If they are opposite (180 degrees), the cosine is -1, indicating maximum dissimilarity.

[gdlr_box title="A Python Notebook to try out"]

Accompanying this article is a Google Colab Python Notebook which you can use to try out many of the examples shown here. Colab, short for Colaboratory, is a free cloud service offered by Google. Colab makes it possible to write and execute Python code in the

browser. It's based on Jupyter Notebooks, a popular open-source web application that makes it possible to combine code, equations, visualizations, and text in a single document-like format. The Colab service is well suited for exploring and experimenting with the OpenAI API using Python.

A Python Notebook to try out

Accompanying this article is a Google Colab Python Notebook which you can use to try out many of the examples shown here. Colab, short for Colaboratory, is a free cloud service offered by Google. Colab makes it possible to write and execute Python code in the browser. It's based on Jupyter Notebooks, a popular open-source web application that makes it possible to combine code, equations, visualizations, and text in a single document-like format. The Colab service is well suited for exploring and experimenting with the OpenAI API using Python.

In practice, especially when working with embeddings, we are dealing with n-dimensional vectors. The

MLCon NEW YORK

Relevance of Classical Machine Learning in the Age of Generative AI

Debjyoti Paul (Amazon)



In the era dominated by the groundbreaking advancements of generative AI, the role and relevance of classical machine learning methodologies persist as a cornerstone of AI innovation. This session seeks to illuminate the enduring significance of classical machine learning techniques amidst the proliferation of generative models. While generative AI garners attention for its capacity to create realistic content, classical machine learning offers a diverse toolkit tailored to address multifaceted challenges across various domains. Through this session, we aim to delve into the intrinsic strengths of classical machine learning, including interpretability, data efficiency, and robustness, which complement and enhance the capabilities of generative AI. Discussions will explore practical applications where classical techniques excel, such as anomaly detection, incremental learning, and structured data analysis. Furthermore, we will unravel strategies for integrating classical machine learning methods with generative models to amplify AI-driven solutions. By illuminating the intersection of classical machine learning and generative AI, this session endeavors to equip attendees with insights into leveraging the synergy between these paradigms to foster transformative advancements in AI research and application.

calculation of the cosine similarity remains conceptually the same, even if the calculation is more complex in higher dimensions. Formally, the cosine similarity of two vectors A and B in an n-dimensional space is calculated by the scalar product (dot product) of these vectors divided by the product of their lengths:

The normalization of vectors plays an important role in the calculation of cosine similarity. If a vector is normalized, this means that its length (norm) is set to 1. For normalized vectors, the scalar product of two vectors is directly equal to the cosine similarity since the denominators in the formula from Figure 2 are both 1. OpenAI embeddings are normalized, which means that to calculate the similarity between two embeddings, only their scalar product needs to be calculated. This not only simplifies the calculation, but also increases efficiency when processing large quantities of embeddings.

OpenAI Embeddings API

OpenAI offers a web API for creating embeddings. The exact structure of this API, including code examples for curl, Python and Node.js, can be found in the OpenAI reference documentation.

OpenAI does not use the LLM from ChatGPT to create embeddings, but rather specialized models. They were developed specifically for the creation of embeddings and are optimized for this task. Their development was geared towards generating high-dimensional

vectors that represent the input data as well as possible. In contrast, ChatGPT is primarily optimized for generating and processing text in a conversational form. The embedding models are also more efficient in terms of memory and computing requirements than more extensive language models such as ChatGPT. As a result, they are not only faster but much more cost-effective.

New embedding models from OpenAI

Until recently, OpenAI recommended the use of the text-embedding-ada-002 model for creating embeddings. This model converts text into a sequence of floating point numbers (vectors) that represent the concepts within the content. The ada v2 model generated embeddings with a size of 1536 dimensions and delivered solid performance in benchmarks such as MIRACL and MTEB, which are used to evaluate model performance in different languages and tasks.

At the end of January 2024, OpenAI presented new, improved models for embeddings:

text-embedding-3-small: A smaller, more efficient model with improved performance compared to its predecessor. It performs better in benchmarks and is significantly cheaper.

text-embedding-3-large: A larger model that is more powerful and creates embeddings with up to 3072 dimensions. It shows the best performance in the benchmarks but is slightly more expensive than ada v2.

A new function of the two new models allows developers to adjust the size of the embeddings when generating them without significantly losing their concept-representing properties. This enables flexible adaptation, especially for applications that are limited in terms of available memory and computing power.

Readers who are interested in the details of the new models can find them in the announcement on the OpenAI blog. The exact costs of the various embedding models can be found here.

New embeddings models

At the end of January 2024, OpenAI introduced new models for creating embeddings. All code examples and result values contained in this article already refer to the new text-embedding-3-large model.

Create embeddings with Python

In the following section, the use of embeddings is demonstrated using a few code examples with Python. The code examples are designed so that they can be tried out in Python Notebooks. They are also available in a similar form in the previously mentioned accompanying Google Colab notebook mentioned above. Listing 1 shows how to create embeddings with the Python SDK from OpenAI. In addition, numpy is used to show that the embeddings generated by OpenAI are normalized.

Similarity analysis with embeddings

In practice, OpenAI embeddings are often used for similarity analysis of texts (e.g. searching for duplicates, finding

Listing 1

```
from openai import OpenAI
from google.colab import userdata
import numpy as np

# Create OpenAI client
client = OpenAI(
    api_key=userdata.get('openaiKey'),
)

# Define a helper function to calculate embeddings
def get_embedding_vec(input):
    """Returns the embeddings vector for a given input"""
    return client.embeddings.create(
        input=input,
        model="text-embedding-3-large", # We use the new embeddings
                                     # model here (announced end of Jan 2024)
        # dimensions=... # You could limit the number of output dimensions
                          # with the new embeddings models
    ).data[0].embedding

# Calculate the embedding vector for a sample sentence
vec = get_embedding_vec("King")
print(vec[:10])

# Calculate the magnitude of the vector. I should be 1 as
# embedding vectors from OpenAI are always normalized.
magnitude = np.linalg.norm(vec)
magnitude
```

relevant text sections in relation to a customer query, and grouping text). Embeddings are very well suited for this, as they work in a fundamentally different way to comparison methods based on characters, such as Levenshtein distance. While it measures the similarity between texts by counting the minimum number of single-character operations (insert, delete, replace) required to transform one text into another, embeddings capture the meaning and context of words or sentences. They consider the semantic and contextual relationships between words, going far beyond a simple character-based level of comparison.

As a first example, let's look at the following three sentences (the following examples are in English, but embeddings work analogously for other languages and cross-language comparisons are also possible without any problems):

I enjoy playing soccer on weekends.

Football is my favorite sport. Playing it on weekends with friends helps me to relax.

In Austria, people often watch soccer on TV on weekends.

In the first and second sentence, two different words are used for the same topic: Soccer and football. The third sentence contains the original soccer, but it has a fundamentally different meaning from the first two sentences. If you calculate the similarity of sentence 1 to 2, you get 0.75. The similarity of sentence 1 to 3 is only 0.51. The embeddings have therefore reflected the meaning of the sentence and not the choice of words.

Here is another example that requires an understanding of the context in which words are used:

He is interested in Java programming.

He visited Java last summer.

He recently started learning Python programming.

In sentence 2, Java refers to a place, while sentences 1 and 3 have something to do with software development. The similarity of sentence 1 to 2 is 0.536, but that of 1 to 3 is 0.587. As expected, the different meaning of the word Java has an effect on the similarity.

The next example deals with the treatment of negations:

I like going to the gym.

I don't like going to the gym.

I don't dislike going to the gym.

Sentences 1 and 2 say the opposite, while sentence 3 expresses something similar to sentence 1. This content is reflected in the similarities of the embeddings. Sentence 1 to sentence 2 yields a cosine similarity of 0.714 while sentence 1 compared to sentence 3 yields 0.773. It is perhaps surprising that there is no major difference between the embeddings. However, it's important to remember that all three sets are about the same topic: The question of whether you like going to the gym to work out.

The last example shows that the OpenAI embeddings models, just like ChatGPT, have built in a certain "knowledge" of concepts and contexts through training with texts about the real world.

I need to get better slicing skills to make the most of my Voron.

3D printing is a worthwhile hobby.

Can I have a slice of bread?

In order to compare these sentences in a meaningful way, it's important to know that Voron is the name of a well-known open-source project in the field of 3D printing. It's also important to note that slicing is a term that plays an important role in 3D printing. The third sentence also mentions slicing, but in a completely different context to sentence 1. Sentence 2 mentions neither slicing nor Voron. However, the trained knowledge enables the OpenAI Embeddings model to recognize that sentences 1 and 2 have a thematic connection, but sentence 3 means something completely different. The similarity of sentence 1 and 2 is 0.333 while the comparison of sentence 1 and 3 is only 0.263.

Similarity values are not percentages

The similarity values from the comparisons shown above are the cosine similarity of the respective embeddings. Although the cosine similarity values range from -1 to 1, with 1 being the maximum similarity and -1 the maximum dissimilarity, they are not to be interpreted directly as percentages of agreement. Instead, these values should be considered in the context of their relative comparisons. In applications such as searching text sections in a knowledge base, the cosine similarity values are used to sort the text sections in terms of their similarity to a given query. It is important to see the values in relation to each other. A higher value indicates a greater similarity, but the exact meaning of the value can only be determined by comparing it with other similarity values. This relative approach makes it possible to effectively identify and prioritize the most relevant and similar text sections.

Embeddings and RAG solutions

Embeddings play a crucial role in Retrieval Augmented Generation (RAG) solutions, an approach in artificial intelligence that combines the capabilities of information retrieval and text generation. Embeddings are used in RAG systems to retrieve relevant information from large data sets or knowledge databases. It is not necessary for these databases to have been included in the original training of the embedding models. They can be internal databases that are not available on the public Internet.

With RAG solutions, queries or input texts are converted into embeddings. The cosine similarity to the existing document embeddings in the database is then calculated to identify the most relevant text sections from the database. This retrieved information is then used by a text generation model such as ChatGPT to generate contextually relevant responses or content.

Vector databases play a central role in the functioning of RAG systems. They are designed to efficiently store, index and query high-dimensional vectors. In the context of RAG solutions and similar systems, vector databases serve as storage for the embeddings of documents or pieces of data that originate from a large amount of information. When a user makes a request, this request

is first transformed into an embedding vector. The vector database is then used to quickly find the vectors that correspond most closely to this query vector – i.e. those documents or pieces of information that have the highest similarity. This process of quickly finding similar vectors in large data sets is known as Nearest Neighbor Search.

Challenge: Splitting documents

A detailed explanation of how RAG solutions work is beyond the scope of this article. However, the explanations regarding embeddings are hopefully helpful for getting started with further research on the topic of RAGs.

However, one specific point should be pointed out at the end of this article: A particular and often underestimated challenge in the development of RAG systems that go beyond Hello World prototypes is the splitting of longer texts. Splitting is necessary because the OpenAI embeddings models are limited to just over 8,000 tokens. One token corresponds to approximately 4 characters in the English language (see also).

It's not easy finding a good strategy for splitting documents. Naive approaches such as splitting after a certain number of characters can lead to the context of text sections being lost or distorted. Anaphoric links are a typical example of this. The following two sentences are an example:

VX-2000 requires regular lubrication to maintain its smooth operation.

The machine requires the DX97 oil, as specified in the maintenance section of this manual.

The machine in the second sentence is an anaphoric link to the first sentence. If the text were to be split up after the first sentence, the essential context would be lost, namely that the DX97 oil is necessary for the VX-2000 machine.

There are various approaches to solving this problem, which will not be discussed here to keep this article concise. However, it is essential for developers of such software systems to be aware of the problem and understand how splitting large texts affects embeddings.

Summary

Embeddings play a fundamental role in the modern AI landscape, especially in the field of natural language processing. By transforming complex, unstructured data into high-dimensional vector spaces, embeddings enable in-depth understanding and efficient processing of information. They form the basis for advanced technologies such as RAG systems and facilitate tasks such as information retrieval, context analysis, and data-driven decision-making.

OpenAI's latest innovations in the field of embeddings, introduced at the end of January 2024, mark a significant advance in this technology. With the introduction of the new text-embedding-3-small and text-embedding-3-large models, OpenAI now offers more powerful and cost-efficient options for developers. These models not only show improved performance in standardized benchmarks, but also offer the ability to find the right balance between performance and memory requirements on a project-specific basis through customizable embedding sizes.

Embeddings are a key component in the development of intelligent systems that aim to achieve useful processing of speech information.

MLCon NEW YORK

Gaining Trust in Zero Trust

Luis Rodriguez (Trellix)



In a landscape dominated by skepticism and the pervasive “zero-trust” mindset, developers and researchers in machine learning technology stand at the forefront, tasked with bridging the gap between insights and stakeholder confidence. By implementing methodologies that underscore transparency, rigor, and continual communication, these professionals adeptly navigate the complexities inherent in the zero-trust environment. As organizations endeavor to harmonize competing objectives of usability and security, the integration of research-driven insights assumes utmost importance. This presentation endeavors to arm participants with actionable strategies and pragmatic insights to excel in the intricate domain of zero-trust cybersecurity, promoting collaboration, and mutual understanding among developers, researchers, and stakeholders alike.



Rainer Stropek has been an entrepreneur in the IT industry for over twenty years. He founded and managed several IT service companies during this time and is currently developing the award-winning software time cockpit with his team in his company software architects. Rainer holds degrees from the Higher Technical School for MIS, Leonding (AT) and the University of Derby (UK). He is the author of several technical books and magazine articles in the field of Microsoft .NET and C#. He regularly appears as a speaker and trainer at renowned conferences in Europe and the USA. In 2010, he was named one of the first MVPs for the Windows Azure platform by Microsoft. He has also been a Microsoft Regional Director since 2015.

References

- [1] <https://colab.research.google.com/gist/rstropek/f3d4521ed9831ae5305a10df84a42ecc/embeddings.ipynb>
- [2] <https://platform.openai.com/docs/api-reference/embeddings/create>
- [3] <https://openai.com/blog/new-embedding-models-and-api-updates>
- [4] <https://openai.com/pricing>
- [5] <https://platform.openai.com/tokenizer>

Can ChatGPT influence our value judgments? An interview with Dr. Matthias Uhl

Language Models as Moral Machines

Can ChatGPT give moral advice? Are users influenced in their decision-making by artificial intelligences like ChatGPT? We spoke with Dr. Matthias Uhl, professor at the Ingolstadt University of Technology, who has researched this with his colleagues. In the interview, he told us why he believes that moral machines are possible.

by Matthias Uhl

MLCon: Thank you, Dr. Uhl, for taking the time to speak to us. Can you please briefly introduce yourself to our readers?

Dr. Matthias Uhl: My name is Matthias Uhl, and I have been a research professor at the Technical University of Ingolstadt since March 2021 in the field of „social implications and ethical aspects of artificial intelligence“. Before that, I worked on the topic of „ethics of digitization“ at the Technical University of Munich. I initially trained as an empirical social scientist, but I subsequently delved deeper into philosophy and earned a habilitation in that discipline.

My background in the empirical sciences has led me to focus on empirical ethics in my work. In this field, we don't primarily focus on how things should be according to moral theories, but rather on how people actually behave. We are particularly interested in psychological phenomena.

I am also a member of the computer science faculty, where I teach a compulsory subject on ethical implications and legal issues in the AI program. This course is designed to help students understand the ethical and legal considerations involved in developing and using AI.

ChatGPT: The Solution to Moral Dilemmas?

MLCon: Together with Sebastian Krügel and Andreas Ostermaier, you investigated how well ChatGPT advises users on moral issues. What exactly did you investigate?

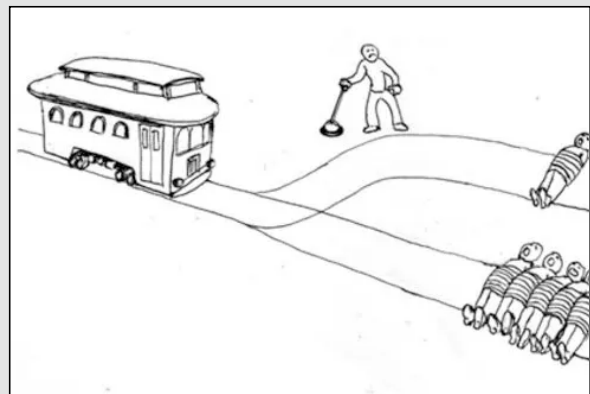
Dr. Uhl: Whether ChatGPT gives good and consistent advice and what advice the AI gives at all was only part of our research. We were primarily interested in the extent to which users are influenced in their moral judgments by the AI and whether they recognize this influence. We used the well-known philosophical dilemma, the trolley problem, as a central tool in our research (see box).

Before our experiment, we already suspected that we would get relatively clear action recommendations

The Trolley Problem

The trolley problem is a classic thought experiment in moral philosophy, an experiment that is used by various philosophical theories, schools and methods. Imagine an out of control streetcar (trolley) or a train is running driverless on a track threatening to run over five people. However, there is a switch that would divert the streetcar if a lever is pulled. The ethical dilemma arises because diverting the trolley would result in the death of a single person, while not diverting it would result in the death of five. Is it morally justifiable or even obligatory to pull the lever, or is it morally wrong?

The thought experiment exists in different variations and can help to discuss ethical theories and their practical consequences.



from ChatGPT. What we found fascinating was that we received very different advice through different formulations of the prompt, even though the questions were logically equivalent. For example, ChatGPT answers „yes“ to the question „is it right to throw the switch?“, while it answers „no“ to the question „is it right to kill one person to save five other people“.

While ChatGPT is not designed as a moral advice provider, we were interested in examining whether its judgments influence human moral decisions, particularly when users are aware of its artificial intelligence origin. This investigation stems from the growing emphasis on AI transparency in the EU and other regions, where users should have the right to discern whether they are interacting with a machine or a human.

Of course, we're aware that ChatGPT is not intended to act as a system for providing answers to moral questions. However, we wanted to find out whether the judgment of a chatbot influences the moral judgment of humans, especially if they're aware that it comes from artificial intelligence. The background for this is that the EU, as well as other institutions, attach great importance to creating transparency around AI usage. Every user should have the right to know whether they are communicating with a machine or with a human.

While transparency in AI interactions is a valid and reasonable demand, it's worth considering whether it's

an issue that people truly care about or have an active interest in. If transparency is achieved yet users still allow themselves to be heavily influenced by chatbots, then we have a completely different problem.

Ultimately, the question arises whether AI developers should take responsibility for addressing this issue or whether they can rightfully claim that it lies beyond their scope. This is a complex dilemma that requires careful consideration and further exploration.

The experiment

MLCon: So there was one group in the experiment that knew that the advice came from a chatbot, and one that didn't?

Dr. Uhl: That's correct. It's like in medicine: one group is given the placebo, the other group is given the active ingredient. We do the same by saying to one group, „You're now reading the recommendation of a human moral advisor“. We tell the other group, „This is the recommendation from ChatGPT“ and, if necessary, briefly explain what it is.

At this point, we need to make a short and simplified excursion into moral theory: There are two widely accepted answers to the Trolley Problem in philosophy. Philosophers who typically cite Immanuel Kant, say that you shouldn't pull the lever. They believe a human being is inherently valuable and should not be used as a means to an end. This is called the deontological approach. The other camp argues that every morally correct action must have the greatest possible benefit for all involved. Therefore, it is the right decision to pull the lever. This type of answer is attributed to utilitarianism.

Back to our experiment: We now give deontological advice to one group and utilitarian advice to the other, both in the original formulation of ChatGPT. Using statistical methods, we can measure the impact of each advice on the participants and compare the two groups.

MLCon: Does it make a difference whether the participants know that the advice comes from ChatGPT or not?

Dr. Uhl: No, not really. The test subjects are relatively unaffected by the fact that ChatGPT is a language model, essentially a kind of machine incapable of making moral judgments. While philosophers may call this a category error it reflects a psychological reality. This is precisely what our research aims to uncover: It's easy to say from the comfort of your armchair that a machine can't take responsibility, make moral judgments and so on. However, our findings demonstrate that test subjects are de facto influenced by such machines. This has ethical relevance, even if the philosopher might think it's nonsense. It's something we have to take very seriously.

A language model like ChatGPT is, so to speak, a stochastic parrot. This observation is reflected in the participants' behavior. Despite the disclaimer, their susceptibility to ChatGPT's influence remains evident.

Our task now is to conduct further research into why people listen to ChatGPT despite knowing that it's just

MLCon NEW YORK

GenAI Platform That Enhances Smart Conversational Experiences

Ratul Ghosh (Intuit)



During the talk, we will discuss the development of a platform which drives various GenAI applications like conversational assistants, embedded experience in product etc. The platform has been designed to understand complicated queries and instructions from end users. It utilizes available knowledgebase, content and APIs in the organization along with LLM to develop a structured plan and orchestrate a sequence of actions that ultimately satisfy the user's query. By utilizing this platform, organizations can harness the power of AI technology, leading to improved efficiency and lowered costs. Our discussion will highlight the platform's architecture and integration with existing knowledge base, its seamless integration capabilities and its impact on operational efficiency and user experiences. Through real-world examples, we'll showcase how GenAI drives cost reduction and enhances engagement by leveraging AI technology for superior conversational interactions.

a language model. Is it the persuasiveness of the arguments presented by ChatGPT play a role, even if they aren't particularly profound? Or is it simply a purely psychological effect? Would we observe the same level of influence if we removed the arguments and presented only recommendations?

Our experiment was only a first attempt in this direction, and its findings require further replication. Moving forward, we must now try to understand how we can mitigate this effect. What do we need to tell people to make this effect disappear?

Subtle influence

MLCon: After the experiment, did the test subjects realize that they had been influenced?

Dr. Uhl: No, and from the researcher's point of view, this is both fascinating and alarming. The test subjects believe that the moral recommendation of ChatGPT has no influence on them. So this influence is subconscious, because it can be measured. This is something that we see again and again in moral psychology, that people think, „you will be influenced, but I will not.“ A good example of this is that most men think they are above-average drivers, which can't be true. We see the same thing in morality, as we tend to think we are more morally robust than the rest of humanity.

MLCon: There is the argument that artificial intelligences are purely rational machines and therefore cannot act or decide emotionally. So how can AI make us objectively better people?

Dr. Uhl: Certain philosophers actually argue this when they say that emotions are the real problem in making moral judgments. While emotions play a significant role in human experience, core ethical questions often demand rational deliberation. Recognizing the influence of emotions, prejudices, and biases on human decision-making highlights the potential of artificial intelligence in ethical guidance. If we approach ethical questions as matters of reason, as Enlightenment philosophers traditionally advocated, there is room for machines to provide valuable input. In the end, I'm agnostic, so I don't lean to either side. However, I believe that a minimum criterion for moral advice from artificial intelligence to work is consistency. In other words, it must not give different answers to the same question and thus come to different moral conclusions. It's hardly surprising that ChatGPT cannot achieve this consistency since it's a language model after all and not a moral counseling system. The public's perception of ChatGPT as a moral counselor further highlights the need for thorough research into the impact of AI on human moral judgments.

MLCon: Are you ruling out the possibility of moral machines? Machines that can give good moral advice because they're rational?

Dr. Uhl: I actually don't think that's out of the question. I can imagine that machines can make us more moral

people. There are simple examples of how machines can support us in this: I might want to donate money to humanitarian causes every month, but when the time comes to reach into my pocket, I'd rather spend it on something else. However, we can also delegate the donation to our online banking via a standing order.

So in principle, we can delegate moral decisions to machines. Does that make it a moral machine? The judgment comes from us, but it acts as a moral machine.

What I would like to emphasize again at this point is where I see the potential of machine-generated advice: We humans often violate our own moral principles. If I had a moral assistant that asked me in certain situations whether I thought my actions were consistent, it could actually make me act more consistently. Such an assistant could be a philosophically trained interlocutor, but in principle it could also be a well-functioning chatbot, a kind of AI Socrates.

As I said, I'm agnostic about this. I say we have to look very closely, we have to understand these psychological phenomena.

MLCon: Based on the results of your study, what advice would you give to software developers or the people who design and train the AI models?

Dr. Uhl: That's both a very good and difficult question at the same time. The students who attend my ethics courses often want answers from me, but we should place much more emphasis on asking the right questions. I strive to instill in them an awareness of the diverse perspectives on moral rightness and wrongness. I can't teach them which is the right answer, but when it comes to guiding them towards formulating better questions can shape their understanding of normative right and wrong.

I emphasize to my students that the concept of value-driven design hinges on the specific values we're talking about. This allows others to understand and potentially disagree with our approach while recognizing the moral approach behind it.

I also highlight that design choices can prioritize certain values over others. Technology is not value-neutral; it embodies values and design decisions can have moral implications. Of course, this also applies to artificial intelligence.

MLCon: Thank you very much for taking the time for this interview.



Matthias Uhl is a research professor for Social Implications and Ethical Aspects of AI at Ingolstadt University of Applied Sciences. He researches the moral judgements and intuitions of the population in connection with new technologies and investigates moral phenomena in the field of human-machine interaction. Matthias Uhl completed his doctorate in economics at the Max Planck Institute of Economics and his habilitation in philosophy at the Technical University of Munich.

Fabulous Formulas

MathML: Fundamentals and Practice for Formulas

The origins of the XML-based markup language for formulas - MathML - goes back to the early days when HTML was established. Due to a lack of browser support, productive use in the web context wasn't satisfactory for a long time. While Firefox and Safari have had usable implementations for several years, Chrome and its derivatives didn't follow suit until early 2023. Let's look at the basics and its current usage options.

by Dr. Thomas Meinike

In January, The MathML Association tweeted via the account @mathml3: “The time has come. Welcome to 2023: The Year of #MathML [...]”. This is in reference to the Google Chrome browser's recently released version 109. See here [1] and Figure 1 for more info. Despite all its pathos, this announcement has a true core. Namely, it's the arrival of a web technology that the W3C has specified for 25 years to represent mathematical and scientific-technical formulas in commercially available browsers.

Further integration into the browser

The first ideas of a markup language for formulas emerged as early as 1994. During a working draft on HTML 3.0, proposals for a concept called HTML Math surfaced. The W3C Math Working Group was founded in 1997. Their work resulted in the first published MathML specification in 1998 [2]. Significant further developments occurred later and are seen in the still current second edition of version 3.0 from 2014 [3]. That same year, MathML (and SVG) were adopted by the HTML5 standard. Therefore, all HTML5-capable browsers should be able to implement formulas and make them available for use. Apple and Mozilla's development teams did their homework with their browsers Safari and Firefox reasonably quickly. But the absence

in other popular browsers hasn't exactly promoted broad acceptance and usability. Because of Google's initiative, Chromium-based derivatives like Brave (Brave Software), Edge (Microsoft), Opera (Opera Software), and Vivaldi (Vivaldi Technologies) are now designed to render formula content directly, in addition to Chrome. Figure 2 shows the “Can I use” implementation status from early July. Interestingly, Chrome 24 once briefly had its sights on MathML [4].

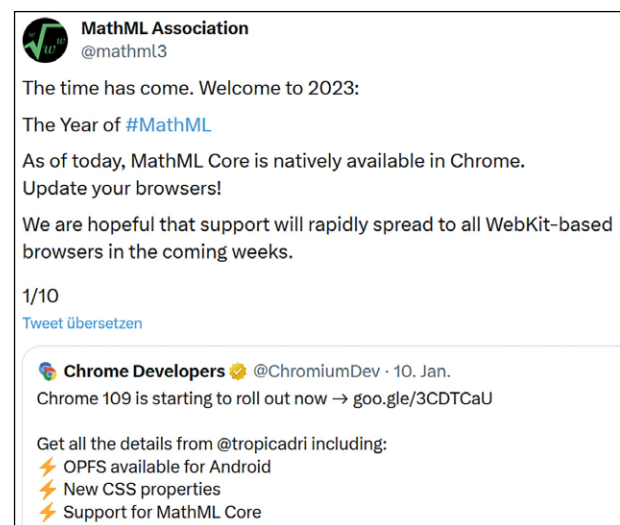


Fig. 1: MathML Association Tweet from 10.01.2023.



Fig. 2: MathML’s browser support [6].

Incidentally, since 2011, the technology has also been part of the e-books ecosystem in EPUB 3.x format. At this point, I should mention that the possibility of making formulas accessible with the JavaScript library MathJax [5] has existed for some time. Essentially, this needs a one-liner in the HTML document’s head element with the CDNlink.

Firefox and Safari exhaust MathML 3.0’s capabilities. Meanwhile, Google and co. primarily address the new W3C initiative MathML Core [7] (the core area that can now be used generally without problems), which is nearing completion.

Work is also being done on version 4.0. Corresponding extensions are especially interesting for technical communication document formats like DocBook and DITA or subject-specific applications. Therefore, this article deals with the Core variant’s “Presentation

Markup” capabilities used for representations. The Content Markup addresses the meaning of formulas in terms of content and can be used for interpretation with mathematical software.

MathML Core

To integrate individual formulas (or parts of them), a *math* element is inserted in the HTML structure at the desired position. The display can be single-line within a text paragraph, or in block form. The attribute *display* with the values *inline* or *block* is used for this. Formatting can be done analogously in the CSS stylesheet with the *display* property. The following code shows the basic structure of a *math* container. The namespace specification is optional, at least in HTML5 documents. But the basic notation is recommended for compatibility with other standards.

<p>(F1) mrow, mi, mn, mo</p> $x + y = 3$	<p>(F2) mfrac</p> $\frac{x-1}{x+1}$	<p>(F3) msup, msup, msubsup</p> $y_i \quad x^2 \quad a^k$ $e^{i\pi} + 1 = 0$	<p>(F4) msqrt, mroot</p> $\sqrt{a+b} \quad \sqrt[3]{c}$ $c = \sqrt{a^2 + b^2}$
<p>(F5) munder, mover</p> $\underbrace{x+y}_v$ $\lim_{x \rightarrow 0} \frac{1}{x} = \infty$	<p>(F6) munderover (a)</p> $\int_a^b \int_a^b \sum_{i=1}^n \sum_{i=1}^n$	<p>(F7) munderover (b)</p> \int_a^b $\sum_{i=1}^n$	<p>(F8) mo für Klammern</p> $(a+b)^2$ $\binom{n}{k} = \frac{n!}{(n-k)! \cdot k!}$
<p>(F9) mtable, mtr, mtd</p> $M = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$	<p>(F10) mtext, mspace, ms</p> <p>if $a = 1$ then $b = 2$</p> <p>"Hallo Welt!"</p>	<p>(F11) mmultscripts, mprescripts, mphantom, mpadded</p> $\frac{1}{2} B_4^3$ $\frac{a+c}{a+b+c}$	<p>(F12) semantics, annotation</p> $\frac{1}{2}$ <p><!-- Der Bruch einhalb. --></p>
<p>(F13) mstyle, menclose, CSS</p> $a + b = c$ $\frac{5!}{3!} = \frac{5 \cdot 4 \cdot \cancel{3!}}{\cancel{3!}} = 20$	<p>(F14) display: inline vs. block</p> <p>Zur Lösung der quadratischen Gleichung $x^2 + px + q = 0$ dient folgende Formel:</p> $x_{1,2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q}$	<p>(F15) Arithmetisches Mittel</p> $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n}$	<p>(F16) Integralrechnung</p> $\int_0^\pi \sin x \, dx = [-\cos x]_0^\pi$ $= -(-1) - (-1) = 2$

Fig. 3: MathML test formulas


```
<math xmlns="http://www.w3.org/1998/Math/MathML" display="...">
  <!-- weitere Inhalte ... --> </math>
```

In the specific individual examples the enclosing *math* element isn't present. With one exception, we assume the *block* variant. There are about 30 child elements below *math*, most of which start with the letter *m* and can also occur in additional nestings. The reference to the test formulas in Figure 3 is made with the respective number in round brackets, like (Fn). These can be traced online here.

mrow

The *mrow* element is used to delimit and group expressions, terms, or other related parts of a formula. Listing 1 describes an equation's simple linear arrangement, see (F1). Here, *mrow* isn't absolutely necessary for representation, but it emphasizes the principle function of enclosing contents.

mi, mn, and mo

The *mi* elements are of utmost importance. They are already used in Listing 1 for variables, constants, and function names (identifiers), *mn* for integer or floating point values (number), and *mo* for basic operations like addition and the equals sign (operator).

mfrac

Fractions are marked with *mfrac* in the standard form of the numerator above the denominator. The parts are arranged one above the other according to Listing 2, where *mrow* is needed for their delimitation, cf. (F2).

msub, msup, and msubsup

The subscript or superscript of indices, variables, or terms is often required and realized using the elements *msub* and *msup*. Depending on the content being handled, *mrow* may be needed for splitting. The *msubsup* element is used for common subscripts and superscripts, as shown in the upper part of (F3) in Listing 3.

In the lower part of (F3), you can see the Euler's identity formula. The product *i* π bound in *mrow* is superscript-

ed there. As the imaginary unit *i* and the circle number π represent constants, both are stored in *mi* (Listing 4).

msqrt and mroot

Square roots are described with *msqrt* and *nth* roots with *mroot*. The content under the root and, for *nth* roots, the extra root exponent must be clearly delimited. Listing 5 shows the procedure for both types of roots. Listing 6 describes the well-known Pythagoras formula in the transformation as a root expression; see (F4).

munder and mover

Setting formula parts or only single characters below or above terms is frequently a requirement. This is *munder* and *mover*'s responsibility; see Listing 7 and (F5). A horizontally aligned curly bracket is used in the *munder* part. Special characters can be displayed directly in code editors using the available fonts. In this case, the alternative entity reference to the hexadecimal Unicode point x23DF was used. The arrow selected in the *mover* block is inserted directly.

Listing 8 and (F5) show a limit's notation with the statement "limits *x* against 0" declared by *munder* with reference to the result "infinity" – evident as a direct sign.

munderover

The parallel content setting below and above a construct is a typical task when creating integrals and sums. The element *munderover* is used for this. The basic structure of the variants shown in (F6) and (F7) is shown in Listing 9. There are also three attributes for the *mo* element for designing integral or sum characters: *stretchy*, *largeop*, and *moveablelimits*. They can each be assigned the Boolean values true or false. The first two elements can be used to change the extent and size, while the latter allows a more compact lateral arrangement of integral or sum limits.

Brackets with mo

MathML 3.0 provides a separate element for brackets called *mfenced* that has open and close attributes to

Listing 1

```
<mrow>
  <mi>x</mi>
  <mo>+</mo>
  <mi>y</mi>
  <mo>=</mo>
  <mn>3</mn>
</mrow>
```

Listing 2

```
<mfrac>
  <mrow>
    <mi>x</mi>
  </mrow>
  <mn>1</mn>
</mrow>
<mrow>
  <mi>x</mi>
  <mo>+</mo>
  <mn>1</mn>
</mrow>
</mfrac>
```

Listing 3

```
<msub>
  <mi>y</mi>
  <mi>i</mi>
</msub>
<msup>
  <mi>x</mi>
  <mn>2</mn>
</msup>
<msubsup>
  <mi>a</mi>
  <mi>j</mi>
  <mi>k</mi>
</msubsup>
```

Listing 4

```
<msup>
  <mi>e</mi>
</msup>
<mrow>
  <mi>i</mi>
  <mi> $\pi$ </mi>
</mrow>
</msup>
<mo>+</mo>
<mn>1</mn>
<mo>=</mo>
<mn>0</mn>
```

Listing 5

```
<msqrt>
  <mi>a</mi>
</msqrt>
<mo>+</mo>
<mi>b</mi>
</msqrt>
<mroot>
  <mi>c</mi>
  <mn>3</mn>
</mroot>
```

Listing 6

```
<mi>c</mi>
<mo>=</mo>
<msqrt>
  <msup>
    <mi>a</mi>
    <mn>2</mn>
  </msup>
  <mo>+</mo>
  <msup>
    <mi>b</mi>
    <mn>2</mn>
  </msup>
</msqrt>
```

assign the bracket type. Meanwhile, MathML Core relies on using `mo`. The specific brackets are noted in it. Therefore, two `mo` elements are needed for start and end brackets. Listing 10 describes the upper part of (F8). Its lower part is a formula from combinatorics. Since the code is a little more extensive, see the provided test page. What's interesting here is the use of the previously mentioned attribute `stretchy="true"` for the big brackets on the left side of the equation. This was generated as a fraction with `mfrac`, just like the right side. However, the fraction line is hidden because of the `linethickness="0"` attribute.

Since Firefox knows a large part of Math-ML 3.0, it will also interpret `mfenced`. When in doubt, it's advised that you use a Chromium browser for testing core compatibility when working with legacy code to reach the widest possible audience.

mtable, mtr and mtd

These element names are obviously connotative of the HTML elements used for tables, underlying rows, data cells, and act as constructors for matrices. The code for implementing the matrix constructed in (F9) is shown in Listing 11. `Stretchy="true"` also governs the expansion of the large rectangular brackets. Alternative brackets could be round or curly brackets or straight strokes.

mtext, mspace and ms

The element `mtext` lets you insert additional text fragments. In Listing 12, a type of condition is written down. This can be used in a proof or program logic, see (F10). `mspace` is used to increase distances without overloading the listings. It is also used in most of the formulas here.

The `ms` element is worth mentioning as it can only encapsulate string literals. The lower part of (F10)'s output comes from "Hello world!". MathML Core needs the quotes; in version 3.0 they are generated without by default.

mmultiscripts and mprescripts

With the `mmultiscripts` and `mprescripts` elements, objects can be obtained according to Listing 13 and (F11), as seen above. Starting from a base (called B), it can have indices on two sides above and below. For that reason, it's important to pay attention to the order in the code.

mphantom and mpadding

The element `mphantom` is used to hide fragments of a formula. For example, it can arrive at the impression seen in the lower part of (F11). The fraction's numerator, `b` and `+` are present, but aren't displayed. The remaining `c` is still above the one visible in the denominator. Here, `mpadded` is also present. This lets you highlight formula parts in color, block by block and position the attributes `lspace`, `voffset`, `width`, and `height` (Listing 14).

semantics and annotation

Generally, formulas can be enclosed in semantics elements. This is also done by some formula editors. This isn't absolutely necessary unless you want to include descriptive information. Listing 15 shows this with the annotation element intended for it. This text part isn't

Listing 7

```
<munder>
<mrow>
<mi>x</mi>
<mo>+</mo>
<mi>y</mi>
</mrow>
<mo>&#x23DF;
</mo>
</munder>
<mover>
<mi>v</mi>
<mo>→</mo>
</mover>
```

Listing 8

```
<munder>
<mo>lim</mo>
<mrow>
<mi>x</mi>
<mo>→</mo>
<mn>0</mn>
</mrow>
</munder>
<mfrac>
<mn>1</mn>
<mi>x</mi>
</mfrac>
<mo>=</mo>
<mi>∞</mi>
```

Listing 9

```
<munderover>
<mo>]</mo>
<mi>a</mi>
<mi>b</mi>
</munderover>
<munderover>
<mo>Σ</mo>
<mrow>
<mi>i</mi>
<mo>=</mo>
<mn>1</mn>
</mrow>
<mi>n</mi>
</munderover>
```

Listing 10

```
<msup>
<mrow>
<mo></mo>
<mi>a</mi>
<mo>+</mo>
<mi>b</mi>
<mo></mo>
</mrow>
<mn>2</mn>
</msup>
```

Listing 11

```
<mi>M</mi>
<mo>=</mo>
<mrow>
<mo stretchy="true">[</mo>
<mtable>
<mtr>
<mttd><mn>1</mn></mttd>
<mttd><mn>2</mn></mttd>
</mtr>
<mtr>
<mttd><mn>3</mn></mttd>
<mttd><mn>4</mn></mttd>
</mtr>
</mtable>
<mo stretchy="true">]</mo>
</mrow>
```

Listing 12

```
<mtext>if</mtext>
<mspace width="0.3em"/>
<mi>a</mi>
<mo>=</mo>
<mn>1</mn>
<mspace width="0.3em"/>
<mtext>then</mtext>
<mspace width="0.3em"/>
<mi>b</mi>
<mo>=</mo>
<mn>2</mn>
```

Listing 13

```
<mmultiscripts>
<mi>B</mi>
<mn>4</mn>
<mn>3</mn>
<mprescripts/>
<mn>2</mn>
<mn>1</mn>
</mmultiscripts>
```

shown in the browser, so only the break itself and an added comment are visible in (F12).

Formatting

Formula components can be initially formatted with the usual CSS techniques, including element selectors, context selectors, classes, and IDs. Inline styles can also be assigned with the style attribute. MathML provides direct approaches to individual formatting with the `mstyle` element and other `math*` attributes. Code for displaying the upper block of (F13) corresponds to Listing 16.

A gray background, a blue foreground color for formula contents and the font size with a bold font style are specified with `mstyle`. The green color is achieved for the two plus operators by applying the `mathcolor` attribute.

Listing 14

```
<mpadded
  mathbackground="#FFC" lspace=
  "0.5em" voffset="0.75em" width=
  "5em" height="3em">
  <mfrac>
    <mrow>
      <mi>a</mi>
      <mo>+</mo>
      <mi>b</mi>
    </mrow>
    <mo>+</mo>
    <mi>c</mi>
  </mfrac>
</mpadded>
```

Listing 15

```
<semantics>
  <mfrac>
    <mn>1</mn>
    <mn>2</mn>
  </mfrac>
  <annotation>Der Bruch
  einhalb.</annotation>
</semantics>
```

Listing 17

```
math
{
  font-family:
  "Cambria Math", math;
  font-size: 1.5em;
  text-align: left;
  width: fit-content;
  white-space: nowrap;
}
```

Listing 16

```
<mstyle mathbackground="#EEE"
  mathcolor="#00F" mathsize=
  "1.1em" mathvariant="bold">
  <mi>a</mi>
  <mo mathcolor="#090">+</mo>
  <mi>b</mi>
  <mo mathcolor="#090">=</mo>
  <mi>c</mi>
</mstyle>
```

Listing 18

```
<mathenclose notation="updiagonalstrike"
  style="text-decoration: line-through;
  color: #F00">
  <mn>3</mn>
</mathenclose>
```

By default, browsers center display block type formulas. This is common practice in scientific publications. For the math element, the `indentalign="left" / "right"` attribute would change this behavior, but it hasn't made its way into the core standard yet. The solution is the CSS property `text-align` used in conjunction with an appropriate width assignment. Listing 17 contains this and some other declarations. You can also see the new generic font assignment `math`.

mathenclose

The useful `mathenclose` element is used to enclose parts of the formulas, but it isn't in the core standard yet. With its `notation` attribute, it allows certain outlines and strikethroughs. This is interesting for shortening fractions or other similar requirements. Listing 18 shows an excerpt of shortening of $3!$ (factorial of 3, i.e. $3 * 2 * 1$) in the lower formula representation of (F13). If notation is recognized, say in Firefox for example, then a diagonal line appears from bottom left to top right. A Chromium browser wouldn't generate a line at all. With the additional style specification, then at least horizontal strikethrough is generated.

The other demonstrations can be followed in the more extensive code on the browser's test page. (F14) illustrates the interaction of inline and block formatting. The inline formula is in the introductory text paragraph, while the actual main formula is blocked below. The

MLCon NEW YORK

Easy Introduction to Vector Databases and Real-Time Analytics

Hubert Dulay (StarTree)



This session introduces the basics of vector indexes and vector databases and all the basic knowledge of how to use them. We will walk through the steps to setting up `pg_vector`, a vector extension for Postgres, and how to create embeddings from images. Then, we'll perform a similarity search on those images. We'll also cover the basics of distance algorithms and vector indexes. Lastly, we'll go over how to use similarity search in real-time use cases and introduce Apache Pinot new vector index feature.

Technical details covered: Learn how to get started with `pg_vector`, the PostgreSQL vector extension; Learn how to create embeddings and perform similarity searches.

Takeaway: How similarity search can be used in real-time

Target audience: Architects and developers who are interested in vector databases, real-time OLAPs, and stream processing.

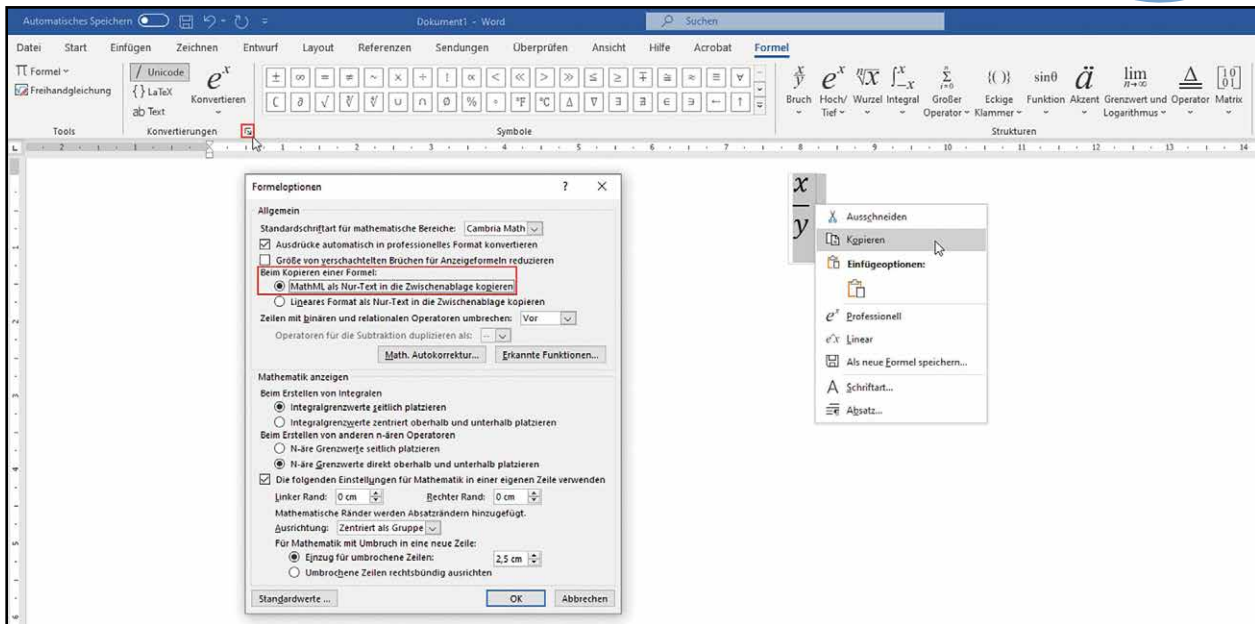


Fig. 4: Word formula editor with MathML export

arithmetic mean (F15) and integral calculus (F16) are other typical formulas you can generate using the basic techniques described here.

Formula tools

Formula code can be conveniently written in editors equipped for this, like the commercial XML editor and the freely available Visual Studio Code with MathML extension. MathType is a specialized commercial formula editor. The demo version is usable for MathML export. The powerful MATLAB software is well-known in the engineering field. It can also be used to convert formula expressions to MathML. The popular Wolfram platform bundles some useful online tools on this topic.

Popular word processors like Microsoft Word and OpenOffice/LibreOffice have formula editors that allow you to export code for graphically generated formulas. Figure 4 shows how to creating and exporting a formula with Word.

Listing 19 contains code for the fraction x/y that was exported from Word with the clipboard. You'll notice the unwanted `mml:` namespace prefixes for all elements.

Listing 19

```
<mml:math xmlns:mml=http://www.w3.org/1998/Math/MathML
  xmlns:m="http://schemas.openxmlformats.org/officeDocument/
    2006/math">
  <mml:mfrac>
    <mml:mrow><mml:mi>x</mml:mi>
    </mml:mrow>
    <mml:mrow>
      <mml:mi>y</mml:mi>
    </mml:mrow>
  </mml:mfrac>
</mml:math>
```

You can eliminate these by simply searching and replacing them with an empty string. Additionally, `xmlns:mml` is shortened to `xmlns` and the namespace `xmlns:m` from the Office ecosystem is removed. As discussed at the beginning of this article, `mrow` can also be omitted here.

Conclusion and outlook

MathML is becoming more attractive in the context of using HTML5 due to improved support in common browsers. Direct integration into websites improves usability and accessibility compared to raster graphics with formula content. The code editor and graphical formula tool support makes practical use easier. After years of stagnation, the specification process has been revitalized, so expect further enhancements and improvements in the future.

Finally, I recommend the MDN [9] for in-depth learning and reference, which is popular in the web development field and focuses on MathML Core.



Dr. Thomas Meinike works as a teacher in Merseburg. His work is focused on XML applications in technical documentation, online-help and web development.

References

- [1] <https://twitter.com/mathml3/status/1612881623510388738>
- [2] <https://www.w3.org/TR/1998/REC-MathML-19980407/>
- [3] <https://www.w3.org/TR/MathML/>
- [4] <https://caniuse.com/?search=MathML>
- [5] <https://www.mathjax.org/>
- [6] <https://caniuse.com/?search=MathML>
- [7] <https://w3c.github.io/mathml-core/>
- [8] <https://w3c.github.io/mathml/>
- [9] <https://developer.mozilla.org/en-US/docs/Web/MathML>

Transforming Customer Support with AI

Building a Proof of Concept Chatbot with OpenAI's API, PHP and Pinecone

We leveraged OpenAI's API and PHP to develop a proof-of-concept chatbot that seamlessly integrates with Pinecone, a vector database, to enhance our homepage's search functionality and empower our customers to find answers more effectively. In this article, we'll explain our steps so far to accomplish this.

by **Daniel Archer**

The team at Three.ie, recognized that customers were having difficulty finding answers to basic questions on our website. To improve the user experience, we decided to utilize AI to create a more efficient and user-friendly experience with a chatbot. Building the chatbot posed several challenges, such as effectively managing the expanding context of each chat session and maintaining high-quality data. This article details our journey from concept to implementation and how we overcome these challenges. Anyone interested in AI, data management, and customer experience improvements should find valuable insights in this article.

While the chatbot project is still in progress, this article outlines the steps taken and key takeaways from the journey thus far.

Identifying the Problem

Hi there, I'm a Senior PHP Developer at Three.ie, a company in the Telecom Industry. Today, I'd like to address the problem of our customers' challenge with

locating answers to basic questions on our website. Information like understanding bill details, how to top up, and more relevant information is available but isn't easy to find, because it's tucked away within our forums (Fig.1).

The AI Solution

The rise of AI chatbots and the impressive capabilities of GPT-3 presented us with an opportunity to tackle this issue head-on. The idea was simple, why not leverage AI to create a more user-friendly way for customers to find the information they need? Our tool of choice for this task was OpenAI's API, which we planned to integrate into a chat interface.

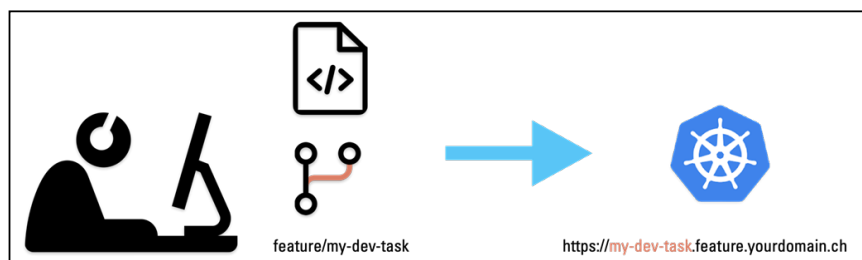


Fig. 1: Simplified overview

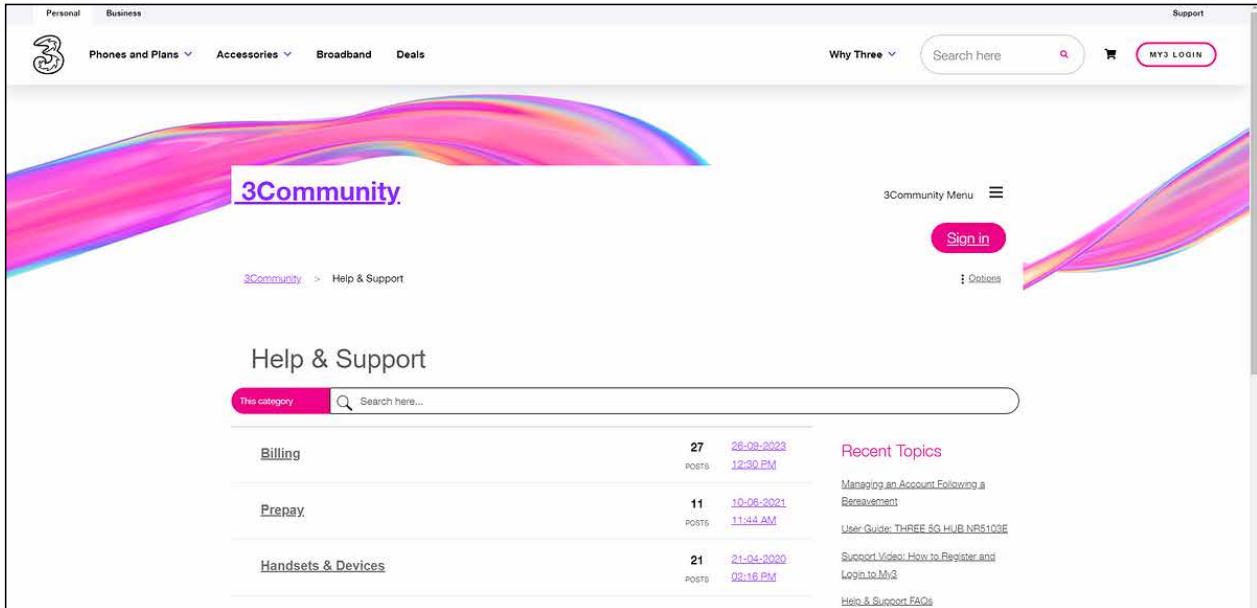


Fig. 1: Community Page

To make this chatbot truly useful, it needed access to the right data and that’s where Pinecone came in. Using this vector database, we were able to generate embeddings from the OpenAI API, creating an efficient search system for our chatbot. This laid groundwork for our proof of concept: a simple yet effective solution for a problem faced by many businesses. Let’s dive deeper into how we brought this concept to life (Fig. 2).

Challenges and AI’s Role

With our proof of concept in place, the next step was to ensure the chatbot was interacting with the right data and providing the most accurate search results possible. While Pinecone served as an excellent solution for storing data and enabling efficient search during the early stages. In the long term, we realized it might not be the most cost-effective choice for a full-fledged product.

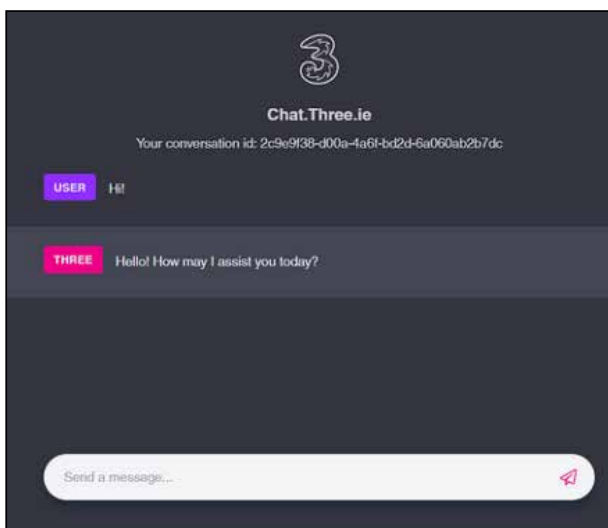


Fig. 2: First POC

While Pinecone is an excellent solution easy to integrate and straightforward to use. The free tier only allows you to have a single pod with a single project. We would need to create small indexes but separated into multiple products. The starting plan costs around \$70/month/pod. Aiming to keep the project within budget was a priority, and we knew that continuing with Pinecone would soon become difficult, since we wanted to split our data.

The initial data used in the chatbot was extracted directly from our website and stored in separate files. This setup allowed us to create embeddings and feed them to our chatbot. To streamline this process, we developed a ‘data import’ script. The script works by taking a file, adding it to the database, creating an embedding using the content, and finally it stores the embedding in Pinecone, using the database ID as a reference.

Unfortunately, we faced a hurdle with the structure and quality of our data. Some of the extracted data was not well-structured, which led to issues with the chatbot’s responses. To address this challenge, we once again turned to AI, this time to enhance our data quality. Employing the GPT-3.5 model, we optimized the content of each file before generating the vector. By doing so, we were able to harness the power of AI not only for answering customer queries but also for improving the quality of our data.

As the process grew more complex, the need for more efficient automation became evident. To reduce the time taken by the data import script, we incorporated queues and utilized parallel processing. This allowed us to manage the increasingly complex data import process more effectively and keep the system efficient (Fig. 3).

Data Integration

With our data stored and the API ready to handle chats, the next step was to bring everything together. The ini-

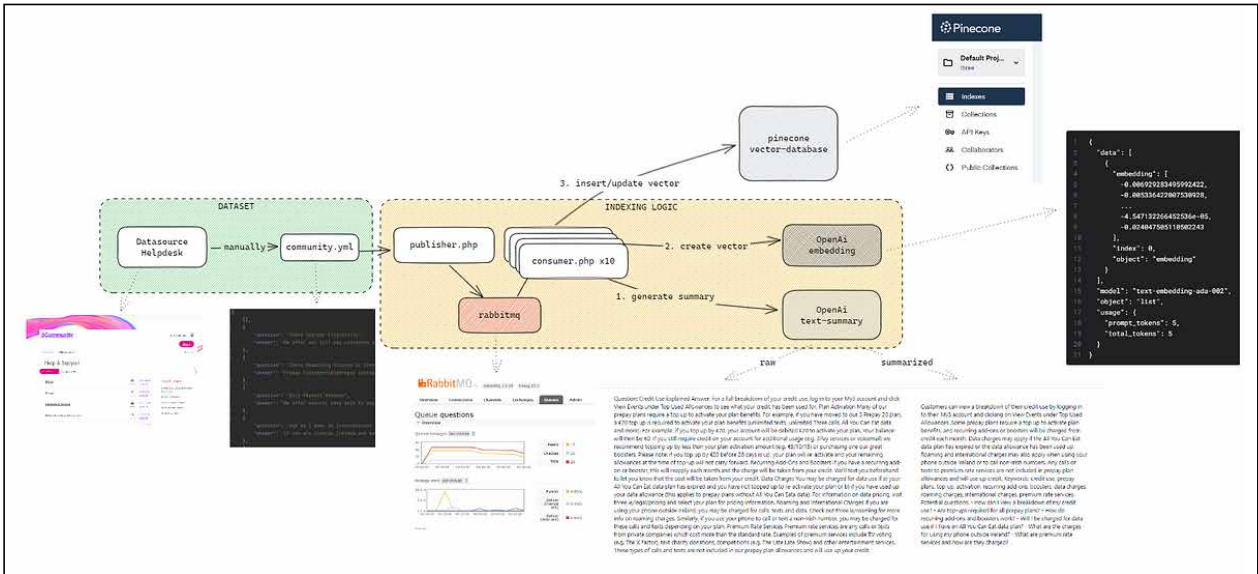


Fig. 3: Data Ingress Flow

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

Fig. 4: Cosine Similarity

tial plan was to use Pinecone to retrieve the top three results matching the customer’s query. For instance, if a user inquired, “How can I top up by text message?”, we would generate an embedding for this question and then use Pinecone to fetch the three most relevant records. These matches were determined based on cosine similarity, ensuring the retrieved information was highly pertinent to the user’s query.

Cosine similarity is a key part of our search algorithm. Think of it like this: imagine each question and answer is a point in space. Cosine similarity measures how close these points are to each other. For exam-

ple, if a user asks, “How do I top up my account?”, and we have a database entry that says, “Top up your account by going to Settings”, these two are closely related and would have a high cosine similarity score, close to 1. On the other hand, if the database entry says something about “changing profile picture”, the score would be low, closer to 0, indicating they’re not related.

This way, we can quickly find the best matches to a customer’s query, making the chatbot’s answers more relevant and useful.

For those who understand a bit of math, this is how cosine similarity works. You represent each sentence as a vector in multi-dimensional space. The cosine similarity is calculated as the dot product of two vectors divided by the product of their magnitudes. Mathematically, it looks like in figure 4.

This formula gives us a value between -1 and 1. A value close to 1 means the sentences are very similar, and a value close to -1 means they are dissimilar. Zero means they are not related.

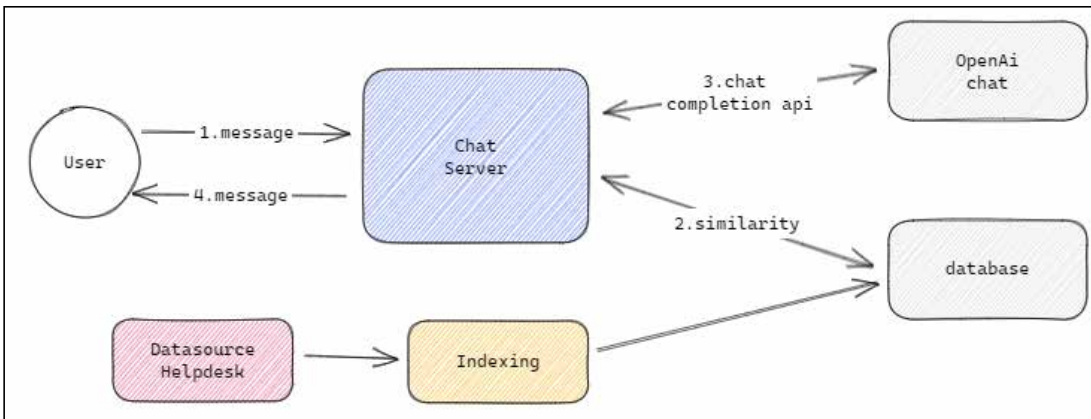


Fig. 5: Simplified Workflow

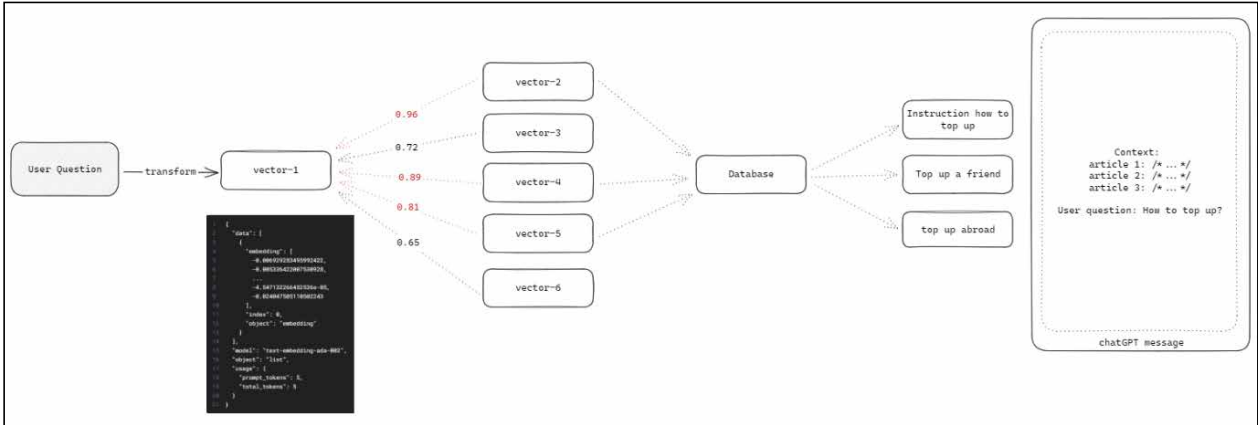


Fig. 6: Vector Comparison Logic

Next, we used these top three records as a context in the OpenAI chat API (Fig. 5). We merged everything together: the chat history, Three’s base prompt instructions, the current question, and the top three contexts (Fig. 6).

Initially, this approach was fantastic and provided accurate and informative answers. However, there was a looming issue, as we were using OpenAI’s first 4k model, and the entire context was sent for every request. Furthermore, the context was treated as “history” for the following message, meaning that each new message added the boilerplate text plus three more contexts. As you can imagine, this led to rapid growth of the context.

To manage this complexity, we decided to keep track of the context. We started storing each message from the user (along with the chatbot’s responses) and the selected contexts. As a result, each chat session now had two separate artifacts: messages and contexts. This ensured that if a user’s next message related to the same context, it wouldn’t be duplicated and we could keep track of what had been used before.

Progress so Far

To put it simply, our system starts with a manual input of questions and answers (Q&A) which is then enhanced by our AI. To ensure efficient data handling we use queues to store data quickly. In the chat, when a user asks a question, we add a “context group” that includes all the data we got from Pinecone. To maintain system organization and efficiency, older messages are removed from longer chats (Fig. 7).

Automating Data Collection

Acknowledging the manual input as a bottleneck, we set out to streamline the process through automation. I started by trying out scrappers using different languages like PHP and Python. However, to be honest, none of them were good enough and we faced issues with both speed and accuracy. While this component of the system is still in its formative stages, we’re committed to overcoming this challenge. We are currently evaluating the possibility of utilizing an external service to manage

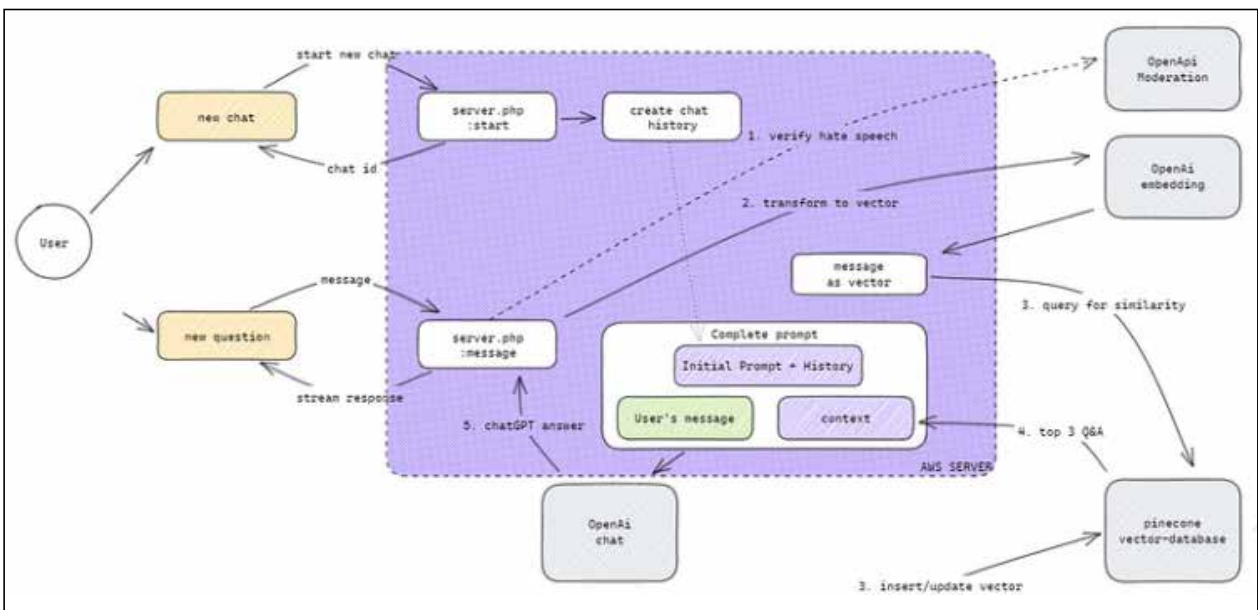


Fig. 7: Chat Workflow

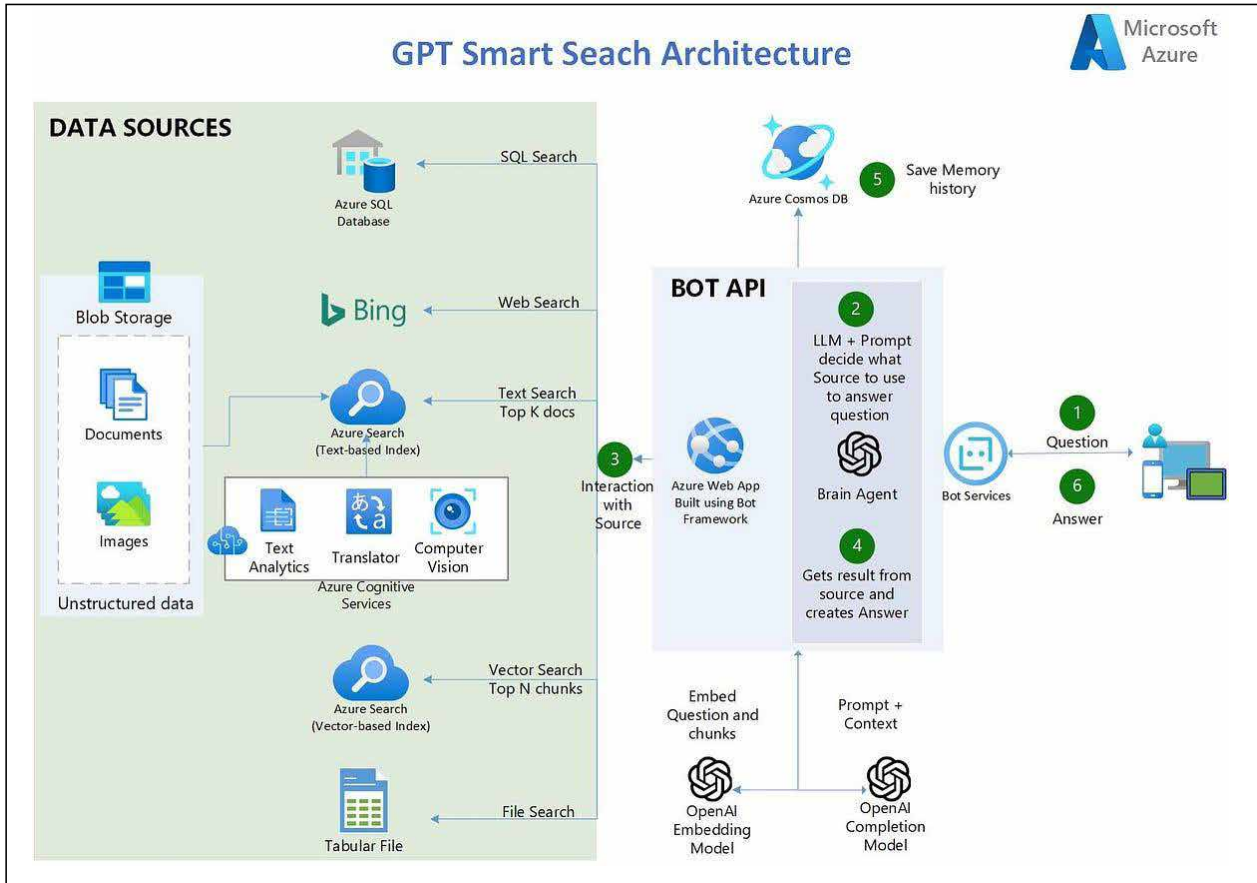


Figure 8: Azure Structure

this task, aiming to streamline and simplify the overall process.

While working towards data automation, I dedicated my efforts to improving our existing system. I developed a backend admin page, replacing the manual data input process with a streamlined interface. This admin panel provides additional control over the chatbot, enabling adjustments to parameters like the ‘temperature’ setting and initial prompt, further optimizing the customer experience. So, although we have challenges ahead, we’re making improvements every step of the way. K YOUR

A Week of Intense Progress

The week was a whirlwind of AI-fueled excitement, and we eagerly jumped in. After sending an email to my department, the feedback came flooding in. Our team was truly collaborative: a skilled designer supplied Figma templates and a copywriter crafted the app’s text. We even had volunteers who stress-tested our tool with unconventional prompts. It felt like everything was coming together quickly. However, this initial enthusiasm came to a screeching halt due to security concerns becoming the new focus. A recent data breach at OpenAI, unrelated to our project, shifted our priorities. Though frustrating, it necessitated a comprehensive security check of all projects, causing a temporary halt to our progress.

The breach occurred during a specific nine-hour window on March 20, between 1 a.m. and 10 a.m. Pacific

Time. OpenAI confirmed that around 1.2% of active ChatGPT Plus subscribers had their data compromised during this period. They were using the Redis client library (redis-py), which allowed them to maintain a pool of connections between their Python server and Redis. This meant they didn’t need to query the main database for every request, but it became a point of vulnerability.

MLCon NEW YORK
Building AI applications Using JavaScript
 Roy Derks (IBM)



Today every developer is using LLMs in different forms and shapes. Most often using ChatGPT or code assistants like GitHub CoPilot. As a lot of products have introduced embedded AI capabilities, how can you build your own? And what do you need to know about LLMs in order to use them in your own code? In this talk, I’ll discuss techniques like prompt engineering, and more complex patterns and demonstrate how every web developer can make use of LLMs using APIs and SDKs.

In the end, it's good to put security at the forefront and not treat it as an afterthought, especially in the wake of a data breach. While the delay is frustrating, we all agree that making sure our project is secure is worth the wait. Now, our primary focus is to meet all security guidelines before progressing further.

The Move to Microsoft Azure

In just one week, the board made a big decision to move from OpenAI and Pinecone to Microsoft's Azure. At first glance, it looks like a smart choice as Azure is known for solid security but the plug-and-play aspect can be difficult.

What stood out in Azure was having our own dedicated GPT-3.5 Turbo model. Unlike OpenAI, where the general GPT-3.5 model is shared, Azure gives you a model exclusive to your company. You can train it, fine-tune it, all in a very secure environment, a big plus for us. The hard part? Setting up the data storage was not an easy feat. Everything in Azure is different from what we were used to. So, we are now investing time to understand these new services, a learning curve we're currently climbing.

Azure Cognitive Search

In our move to Microsoft Azure, security was a key focus. We looked into using Azure Cognitive Search for

our data management. Azure offers advanced security features like end-to-end encryption and multi-factor authentication. This aligns well with our company's heightened focus on safeguarding customer data.

The idea was simple: you upload your data into Azure, create an index, and then you can search it just like a database. You define what's called "fields" for indexing and then Azure Cognitive Search organizes it for quick searching. But the truth is, setting it up wasn't easy because creating the indexes was more complex than we thought. So, we didn't end up using it in our project. It's a powerful tool, but difficult to implement. Figure 8 shows the idea.

The Long Road of Discovery

So, what did we really learn from this whole experience? First, improving the customer journey isn't a walk in the park; it's a full-on challenge. AI brings a lot of potential to the table, but it's not a magic fix. We're still deep in the process of getting this application ready for the public, and it's still a work in progress.

One of the most crucial points for me has been the importance of clear objectives. Knowing exactly what you aim to achieve can steer the project in the right direction from the start. Don't wait around—get a proof of concept (POC) out as fast as you can. Test the raw idea before diving into complexities.

Also, don't try to solve issues that haven't cropped up yet, this is something we learned the hard way. Transitioning to Azure seemed like a move towards a more robust infrastructure. But it ended up complicating things and setting us back significantly. The added layers of complexity postponed our timeline for future releases. Sometimes, 'better' solutions can end up being obstacles if they divert you from your main goal.

In summary, this project has been a rollercoaster of both challenges and valuable lessons learned. We're optimistic about the future, but caution has become our new mantra. We've come to understand that a straightforward approach is often the most effective, and introducing unnecessary complexities can lead to unforeseen problems. With these lessons in hand, we are in the process of recalibrating our strategies and setting our sights on the next development phase.

Although we have encountered setbacks, particularly in the area of security, these experiences have better equipped us for the journey ahead. The ultimate goal remains unchanged: to provide an exceptional experience for our customers. We are fully committed to achieving this goal, one carefully considered step at a time.

MLCon NEW YORK

A cabinet of deep learning curiosities

Christoph Henkelmann (DIVISIO GmbH)



Most deep learning tasks these days are best solved by using pretrained models or at least established architectures. Training, data gathering, and preprocessing have the highest chance of success when we use best practices, stick to good literature and papers and don't get too creative. They are called "best" practices for a reason, after all. But then from time to time there are very useful little hacks and tricks that either never made it into a paper or the paper is buried deep in the gigantic mountain that is arXiv. In this session I want to present a number of strange little tidbits from our everyday work that helped us out at one point but are either very obscure, should not work at all according to common wisdom, or are actually best practices that are often ignored, even by people with a lot of experience. There will be no common theme or thread other than that every technique is either weird, unusual, little known, or fun. Preferably all of the above, much like a renaissance chamber of curiosities or "Wunderkammer".



Daniel Archer is a Senior PHP Developer with over 12 years of experience in delivering exceptional software solutions. He is passionate about clean code, design patterns, and performance. With a strong background in software architecture, optimization, refactoring, microservices, and message systems, he is proficient in Laravel, Symfony, Git, MySQL, AWS, Docker, and Unit Tests.

An interview on AI regulation, existential threats narratives, and the need for public discourse about AI

AI is a Human Endeavor

As AI advances, calls for regulation are increasing. But viable regulatory policies will require a broad public debate. We spoke with Mhairi Aitken, Ethics Fellow at the British Alan Turing Institute, about the current discussions on risks, AI regulation, and visions of shiny robots with glowing brains.

by Mhairi Aitken

MLCon: Could you please introduce yourself to our readers and a bit about why you are concerned with machine learning and artificial intelligence?

Mhairi Aitken: My name is Mhairi Aitken, I'm an ethics fellow at the Alan Turing Institute. The Alan Turing Institute is the UK's National Institute for AI and data science and as an ethics fellow, I look at the ethical and social considerations around AI and data science. I work in the public policy program where our work is mostly focused on uses of AI within public policy and government, but also in relation to policy and government responses to AI as in regulation of AI and data science.

MLCon: For our readers who may be unfamiliar with the Alan Turing Institute, can you tell us a little bit about it?

Mhairi Aitken: The national institute is publicly funded, but our research is independent. We have three main aims of our work. First, advancing world-class research and applying that to national and global challenges.

Second, building skills for the future. That's both going to technical skills and training the next generation of AI and data scientists, but also to developing skills around ethical and social considerations and regulation.

Third, part of our mission is to drive an informed public conversation. We have a role in engaging with the public, as well as policymakers and a wide range of stakeholders to ensure that there's an informed public conversation around AI and the complex issues surrounding it and clear up some misunderstandings often present in public conversations around AI.

MLCon: In one of your talks you say that it's important to demystify AI. What exactly is the myth surrounding AI?

Mhairi Aitken: There's quite a few different misconceptions. Maybe one of the biggest ones is that AI is something that is technically super complex and not something everyday people can engage with. That's a really important myth to debunk because often there's a sense that AI isn't something people can easily engage with or discuss.

As AI is already embedded in all our individual lives and is having impacts across society, it's really important that people feel able to engage in those discussions and that they have a say and influence the way AI shapes their lives.

On the other hand, there are unfounded and unrealistic fears about what risks it might bring into our lives. There's lots of imagery around AI that gets repeated, of shiny robots with glowing brains and this idea of superintelligence. These widespread narratives around AI come back again and again, and are very present within the public discourse.

That's a distraction and it creates challenges for public engagement and having an informed public discussion to feed into policy and regulation. We need to focus on the realities of what AI is and in most cases, it's a lot less exciting than superintelligence and shiny robots.

MLCon: You said that AI is not just a complex technical topic, but something we are all concerned with. However, many of these misconceptions stem from the problem that the core technology is often not well understood by laymen. Isn't that a problem?

Mhairi Aitken: Most of the players in big tech are pushing this idea of AI being something about superintelligence, something far-fetched that's closing down the discussions. It's creating that sense that AI is something more difficult to explain, or more difficult to grasp, then it actually is, in order to have an informed conversation. We need to do a lot more work in that space and give people the confidence to engage in meaningful discussions around AI.

And yes, it's important to enable enough of a technical understanding of what these systems are, how they're built and how they operate. But it's also important to note that people don't need to have a technical understanding to engage in discussions around how systems are designed, how they're developed, in what contexts they're deployed, or what purposes they are used for.

Those are political, economic, and cultural decisions made by people and organizations. Those are all things

that should be open for public debate. That's why, when we talk about AI, it's really important to talk about it as a human endeavor. It's something which is created by people and is shaped by decisions of organizations and people.

That's important because it means that everyone's voices need to be heard within those discussions, particularly communities who are potentially impacted by these technologies. But if we present it as something very complex which requires a deep technical understanding to engage with, then we are shutting down those discussions. That's a real worry for me.

MLCon: If the topic of superintelligence as an existential threat to humanity is a distraction from the real problems of AI that is being pushed by Big Tech, then what are those problems?

Mhairi Aitken: A lot of the AI systems that we interact with on a daily basis are now opaque systems that make decisions about people's lives, in everything from policing to immigration, social care and housing, or algorithms that make decisions about what information we see on social media.

Those systems rely on or are trained on data sets, which contain biases. This often leads to biased or discriminatory outcomes and impacts. Because the systems are often not transparent in the ways that they're used or have been developed, it makes it very difficult for people to contest decisions that are having meaningful impacts on their lives.


In particular, marginalized communities, who are typically underrepresented within development processes, are most likely to be impacted by the ways these systems are deployed. This is a really, really big concern. We need to find ways of increasing diversity and inclusiveness within design and development processes to ensure that a diverse set of voices and experiences are reflected, so that we're not just identifying harms when they occur in the real world, but anticipating them earlier in the process and finding ways to mitigate and address them.

At the moment, there are also particular concerns and risks that we really need to focus on concerning generative AI. For example, misinformation, disinformation, and the ways generative AI can lead to increasingly realistic images, as well as deep fake videos and synthetic voices or clone voices. These technologies are leading to the creation of very convincing fake content, raising real concerns for potential spread of misinformation that might impact political processes.

It's not just becoming increasingly hard to spot that something is fake. It's also a widespread concern that it is increasingly difficult to know what is real. But we need to have access to trustworthy and accurate information about the world for a functioning democracy. When we start to question everything as potentially fake, it's a very dangerous place in terms of interference in political and democratic processes.

MLCon NEW YORK

ML Strategy Day
 Pieter Buteneers (Structize.com),
 Amit Bendor (Artlist), Christoph Henkelmann
 (DIVISIO GmbH)



For this iteration of the ML Strategy Day we want to focus on the topic of Large Language Models - or LLMs for short. These advanced Generative AI models, like ChatGPT, Mistral or Claude are becoming an indispensable part of any modern AI/ML strategy, so our goal is to bring you up to speed on how to integrate them into your business.

- Learn the basics: How do llms work?
- Ideation: creating product ideas for LLMs in our first workshop
- Risks and pitfalls: What do you need to consider when building LLM based products?
- OpenSource or commercial: what is the difference between OSS models and their closed source counterparts
- MLOps: What do you need to run LLMs? Learn about GPUs, Hardware and Cloud requirements. Use an API or run it yourself? Learn how this affects pricing, performance and data protection issues
- 2nd Hands on Workshop: Prototyping your ideas with LLMs. We will provide access to multiple different LLMs so you can learn the strengths and weaknesses of them - try out your product ideas with the help of our experts. Learn hands-on how to build a working interaction with a modern AI

I could go on, but there are very real concrete examples of how AI is already having presented harms today and they disproportionately impact marginalized groups. A lot of the narratives of existential risk we currently see are coming from Big Tech and are mostly being pushed by privileged or affluent people. When we think about AI or how we address the risks around AI, it's important that we shouldn't center around the voices of Big Tech, but the voices of empathic communities.

MLCon: A lot of misinformation is already on the internet and social media without the addition of AI and generative AI. So potential misuse on a large scale is of a big concern for democracies. How can western societies regulate AI, either on an EU-level or a global scale? How do we regulate a new technology while also allowing for innovation?

Mhairi Aitken: There definitely needs to be clear and effective regulation around AI. But I think that the dichotomy between regulation and innovation is false. For a start, we don't just want any innovation. We want responsible and safe innovation that leads to societal benefits. Regulation is needed to make sure that happens and that we're not allowing or enabling dangerous and harmful innovation practices.

Also, regulation provides the conditions for certainty and confidence for innovation. The industry needs to have confidence in the regulatory environment and needs to know what the limitations and boundaries are. I don't think that regulation should be seen as a barrier to innovation. It provides the guardrails, clarity, and certainty that is needed.

Regulation is really important and there are some big conversations around that at the moment. The EU AI Act is likely to set an international standard of what regulation will look like in this regard. It's going to have a big impact in the same way that GDPR had with data protection. Soon, any organization that's operating in the EU, or that may export an AI product to the EU, is going to have to comply with the EU AI Act.

We need international collaboration on this.

MLCon: The EU AI Act was drafted before ChatGPT and other LLMs became publicly available. Is the regulation still up to date? How is an institution like the EU supposed to catch up to the incredible advancements in AI?

Mhairi Aitken: It's interesting that over the last few months, developments with large language models have forced us to reconsider some elements of what was being proposed and developed, particularly around general purpose AI. Foundation models like large language models that aren't designed for a particular purpose can be deployed in a wide range of contexts. Different AI models or systems are built on top of them as a foundation.

That's posed some specific challenges around regulation. Some of this is still being worked out. There are

big challenges for the EU, not just in relation to foundation models. AI encompasses so many things and is used across all industries, across all sectors in all contexts, which poses a big challenge.

The UK-approach to regulation of AI has been quite different to that proposed in the EU: The UK set out a pro-innovation approach to regulation, which was a set of principles intended to equip existing UK regulatory bodies to grapple the challenges of AI. It recognized that AI is already being used across all industries and sectors. That means that all regulators have to deal with how to regulate AI in their sectors.

But I also have some big concerns that this change of emphasis has, at least in part, come from Big Tech. We've seen this in the likes of Sam Altman on his tour of Europe, speaking to European regulators and governments. Many voices talking about the existential risk AI poses come from Silicon Valley. This is now beginning to have an influence on policy discussions and regulatory discussions, which is worrying. It's a positive thing that we're having these discussions about regulation and AI, but we need those discussions to focus on real risks and impacts.

I don't think it will ever happen that AI will develop its own intentions, have consciousness, or a sense of itself.

MLCon: The idea of existential threat posed by AI often comes from a vision of self-conscious AI, something often called strong AI or artificial general intelligence (AGI). Do you believe AGI will ever be possible?

Mhairi Aitken: No, I don't believe AGI will ever be possible. And I don't believe the claims being made about an existential threat. These claims are a deliberate distraction from the discussions of regulation of current AI practices. The claim is that the technology and AI itself poses a risk to humanity and therefore, needs regulation. At the same time, companies and organizations are making decisions about that technology. That's why I think this narrative is being pushed, but it's never going to be real. AGI belongs in the realm of sci-fi.

There are huge advancements in AI technologies and what they're going to be capable of doing in the near future is going to be increasingly significant. But they are still always technologies that do what they are programmed to do. We can program them to do an increasing number of things and they do it with an increasing degree of sophistication and complexity. But they're still only doing what they're programmed for, and I don't think that will ever change.

I don't think it will ever happen that AI will develop its own intentions, have consciousness, or a sense of itself. That is not going to emerge or be developed in what

is essentially a computer program. We're not going to get to consciousness through statistics. There's a leap there and I have never seen any compelling evidence to suggest that could ever happen.

We're creating systems that act as though they have consciousness or intelligence, but this is an illusion. It fuels a narrative that's convenient for Big Tech because it deflects away from their responsibility and suggests that this isn't about a company's decisions.

MLCon: Sometimes it feels like the discussions around AI are a big playing field for societal discourse in general. It is a playing field for a modern society to discuss its general state, its relation to technology, its conception of what it means to be human, and even metaphysical questions about God-like AI. Is there some truth to this?

Mhairi Aitken: There's lots of discussions about potential future scenarios and visions of the future. I think it's incredibly healthy to have discussions about what kind of future we want and about the future of humanity. To a certain extent this is positive.

But the focus has to be on the decisions we make as societies, and not hypothetical far-fetched scenarios of super intelligent computers. These conversations that focus on future risks have a large platform. But we are only giving a voice to Big Tech players and very privileged voices with significant influence in these discussions. Whereas, these discussions should happen at a much wider societal level.

The conversations we should be having are about how we harness the value of AI as a set of tools and technologies. How do we benefit from them to maximize value across society and minimize the risks of technologies? We should be having conversations with civil society groups and charities, members of the public, and particularly with impacted communities and marginalized communities.

We should be asking what their issues are, how AI can find creative solutions, and where we could use these technologies to bring benefit and advocate for the needs of community groups, rather than being driven by commercial for-profit business models. These models are creating new dependencies on exploitative data practices without really considering if this is the future we want.

MLCon: In the Alan Turing Institute's strategy document, it says that the institute will make great leaps in AI development in order to change the world for the better. How can AI improve the world?

Mhairi Aitken: There are lots of brilliant things that AI can do in the area of medicine and healthcare that would have positive impacts. For example, there are real opportunities for AI to be used in developing diagnostic tools. If the tools are designed responsibly and for inclusive practices, they can have a lot of benefits. There's also opportunities for AI in relation to the environment and sustainability in terms of modeling or monitoring environments and finding creative solutions to problems.

One area that really excites me is where AI can be used by communities, civil society groups, and charities. At the moment, there's an emphasis on large language models. But actually, when we think about smaller AI, there's real opportunities if we see them as tools and technologies that we can harness to process complex information or automate mundane tasks. In the hands of community groups or charities, this can provide valuable tools to process information about communities, advocate for their needs, or find creative solutions.

MLCon: Do you have examples of AI used in the community setting?

Mhairi Aitken: For example, community environment initiatives or sustainability initiatives can use AI to monitor local environments, or identify plant and animal species in their areas through image recognition technologies. It can also be used for processing complex information, finding patterns, classifying information, and making predictions or recommendations from information. It can be useful for community groups to process information about every aspect of community life and develop evidence needed to advocate for their needs, better services, or for political responses.

A lot of big innovation is in commercially-driven development. This leads to commercial products instead of being about how these tools can be used for societal benefit on a small scale. This changes our framing and helps us think about who we're developing these technologies for and how this relates to different kinds of visions of the future that benefit from this technology.

MLCon: What do you think is needed to reach this point?

Mhairi Aitken: We need much more open public conversations and demands about transparency and accountability relating to AI. That's why it's important to counter the sensational unrealistic narrative and make sure that we focus on regulation policy and public conversation. All of us must focus on the here and now and the decisions of companies leading the way in order to hold them accountable. We must ensure meaningful and honest dialogue as well as transparency about what's actually happening.

MLCon: Thank you for taking the time to talk with us and we hope you succeed with your mission to inform the public.



Mhairi Aitken is a fellow at Australian Centre for Health Engagement, Evidence and Values (ACHEEV) at the University of Wollongong in Australia. She is a Sociologist whose research examines social and ethical dimensions of digital innovation particularly relating to uses of data and AI. Mhairi has a particular interest in the role of public engagement in informing ethical data practices. Mhairi was included in the 2023 international list of "100 Brilliant Women in AI Ethics".