British Actuarial Journal (2025), Vol. **30**, e7, pp. 1–14
doi:10.1017/S1357321724000187

Institute and Faculty of Actuaries

SESSIONAL MEETING DISCUSSION

ActuaryGPT: applying LLMs to insurance and actuarial work

[Institute and Faculty of Actuaries Sessional Webinar, Wednesday 21 February 2024]

Moderator (Miss R. Rustagi): Good evening everyone. I am your host for today. My name is Ritika Rustagi and I am a Non-Life Risk Director and a Capital Specialist. I am also a member of the IFoA GI Research and Thought Leadership Committee, which supports the GI Board.

Today, we will discuss a paper that was submitted for the Brian Hey Award in 2023 and was Highly Commended. The author, Caesar (Balona), has been invited to present this interesting paper. The recent advances in Large Language Models (LLMs) have spurred interest in their potential applications across various fields, including actuarial work. The paper introduces the use of LLMs in actuarial and insurance-related tasks, both as direct contributors to actuarial modelling and as workflow assistants. It provides an overview of LLM concepts and potential applications in actuarial science and insurance. This webinar will explore the main elements of Caesar's paper: what LLMs are, their uses, two case studies, and the risks that AI brings, as well as the new developments since the paper was written.

I would like to introduce Caesar (Balona). He is a qualified actuary with experience in the short-term consulting industry. Recently, he headed up the Data Science team at QED Actuaries & Consultants. He is currently leading Catastrophe and Climate Modelling at Old Mutual in South Africa, where he uses machine learning and programming to gain insights on climate risk from remote sensing data. A significant part of his work lies at the intersection of actuarial science, software development, and machine learning. He contributes very widely to the broader community through his website, which can be found at https://modernactuary.co.za/.

Mr C. Balona: Thank you all for joining this webinar today. What I want everyone to learn from this webinar is an answer to the question, 'How can we use LLMs?' This webinar is based on a paper I have written. If you would like more detail, please refer to the paper (Balona, 2023).

I will give a brief introduction to LLMs so you have a little bit more knowledge about them. I will also discuss some ways to use them with motivating examples, and finally make closing remarks and talk about future developments.

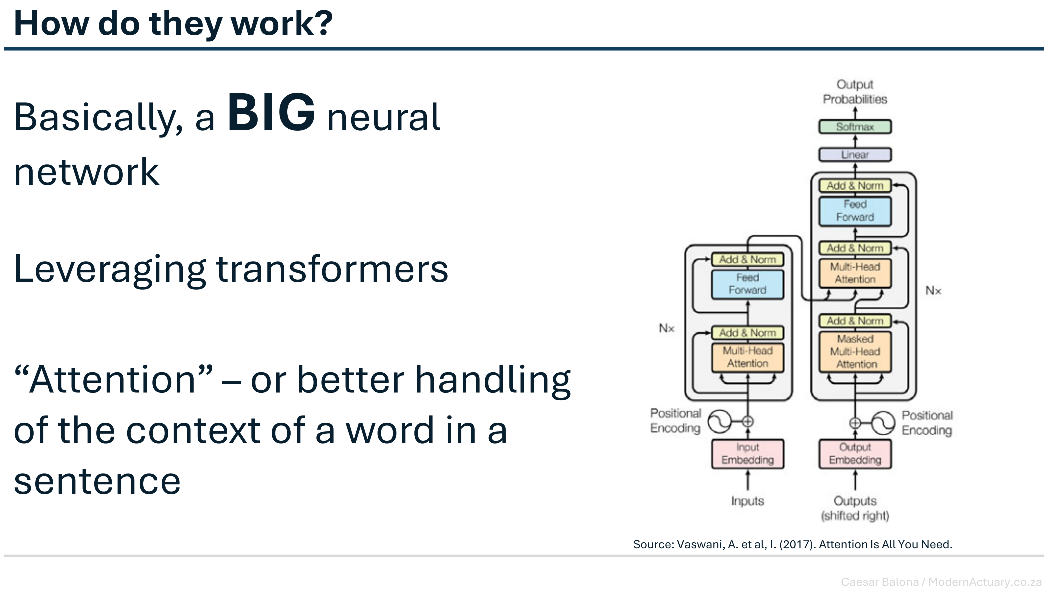
What is a Large Language Model (LLM)? Chances are that you have already been exposed to an LLM through a popular website called ChatGPT. ChatGPT is a user interface on top of an LLM. If you have ever used ChatGPT, you have used an LLM already. There are many other ways to use them and incorporate them into our work.

An LLM is a big neural network. When I say 'big', I mean it is very big. The difference in scale between a supercomputer and a typical calculator is the approximate scale of the difference between an LLM that you use through ChatGPT and a typical neural network that you may use in your day-to-day work.

One of the key breakthroughs of LLMs is that they leverage a transformer. Transformers are different mathematical constructs that are paired together in certain ways and you pass data through them to get a certain output. What makes them well suited to language modelling is

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

something called ‘Attention’, which is better handling of the context of a word within a sentence. These models can understand how a word fits into a sentence and what it might mean.



How do they work?

Basically, a **BIG** neural network

Leveraging transformers

“Attention” – or better handling of the context of a word in a sentence

Source: Vaswani, A. et al, I. (2017). Attention Is All You Need.

Caesar Balona / ModernActuary.co.za

Figure 1. How do LLMs work?

For example, if I say I am talking about transformers in this presentation, you know I am talking about the transformer on the right-hand side of Figure 1. If I was talking about electricity or movies, I would be talking about a different type of transformer. Transformers can understand that context, which makes them powerful for language tasks.

Looking at that image on the right and hearing about supercomputers, you may be thinking ‘Wow, this sounds really complicated’ and ‘Can we utilise it?’ LLMs are in fact very accessible and immediately useful.

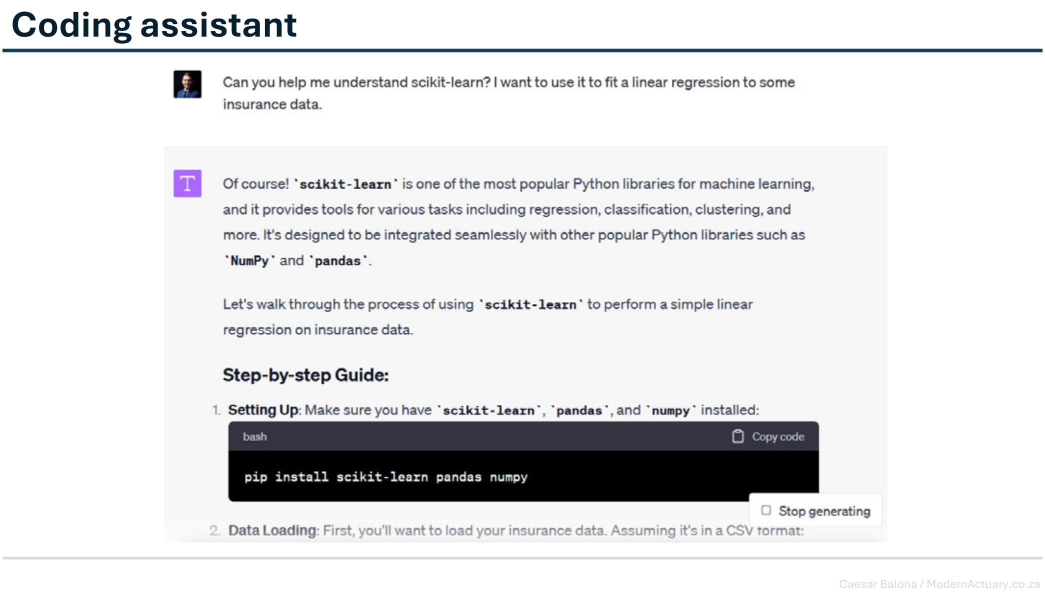
I have shown you ChatGPT already. You can input your prompts and ask a question. The LLM will do its best to help you. There are many other models and websites. Another popular one is Bing by Microsoft, available via the Bing website. I will touch on some of the others towards the end of the presentation. These websites offer easily accessible ways to get informal help from an LLM. What I mean by ‘informal’ is that you are interacting through a website, and it is helping you with your workflow. It is not directly involved in the workflow.

A great example of that is as a coding assistant. Maybe you are an actuary who wants to get involved in coding and you have heard of a Python package called scikit-learn. But you have no idea how to use it. You can go to ChatGPT, or another website, and say, ‘I have heard about scikit-learn. Can you help me understand it? I want to fit a linear regression to some data.’ As you can see in Figure 2, it will put together a step-by-step guide that is tailored to your specific problem.

You can also directly upload data (we will speak about the risks of doing that near the end of the presentation). If you feel comfortable, you can upload some data and tell it directly ‘Here is some data. Can you do some analysis on it?’ It will take the data and write some Python code. It will even run the Python code and show you the results live. So, it has never been easier to try these technologies.

We should pause here and think about one of the most important considerations when interacting with an LLM. It is something called a prompt. A prompt is the instruction you give to the LLM. In the example in Figure 3, I give an example of a bad prompt.

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press



Coding assistant

Can you help me understand scikit-learn? I want to use it to fit a linear regression to some insurance data.

Of course! `scikit-learn` is one of the most popular Python libraries for machine learning, and it provides tools for various tasks including regression, classification, clustering, and more. It's designed to be integrated seamlessly with other popular Python libraries such as `NumPy` and `pandas`.

Let's walk through the process of using `scikit-learn` to perform a simple linear regression on insurance data.

Step-by-step Guide:

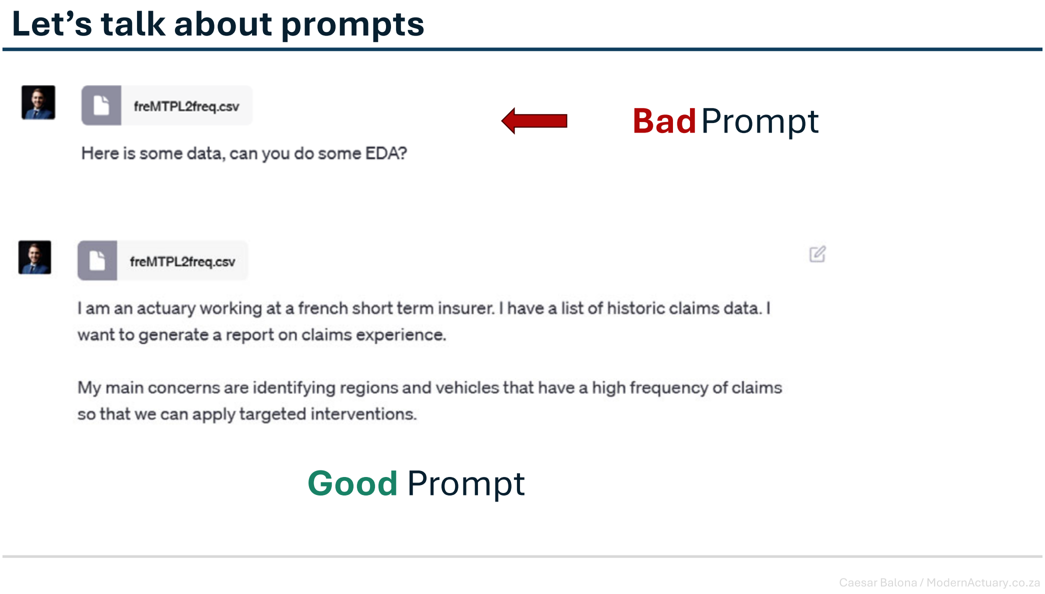
1. **Setting Up**: Make sure you have `scikit-learn`, `pandas`, and `numpy` installed:

```bash
pip install scikit-learn pandas numpy
```

2. **Data Loading**: First, you'll want to load your insurance data. Assuming it's in a CSV format:

Caesar Balona / ModernActuary.co.za

Figure 2. Coding assistant.



Let's talk about prompts

freMTPL2freq.csv

Here is some data, can you do some EDA?

Bad Prompt

freMTPL2freq.csv

I am an actuary working at a french short term insurer. I have a list of historic claims data. I want to generate a report on claims experience.

My main concerns are identifying regions and vehicles that have a high frequency of claims so that we can apply targeted interventions.

Good Prompt

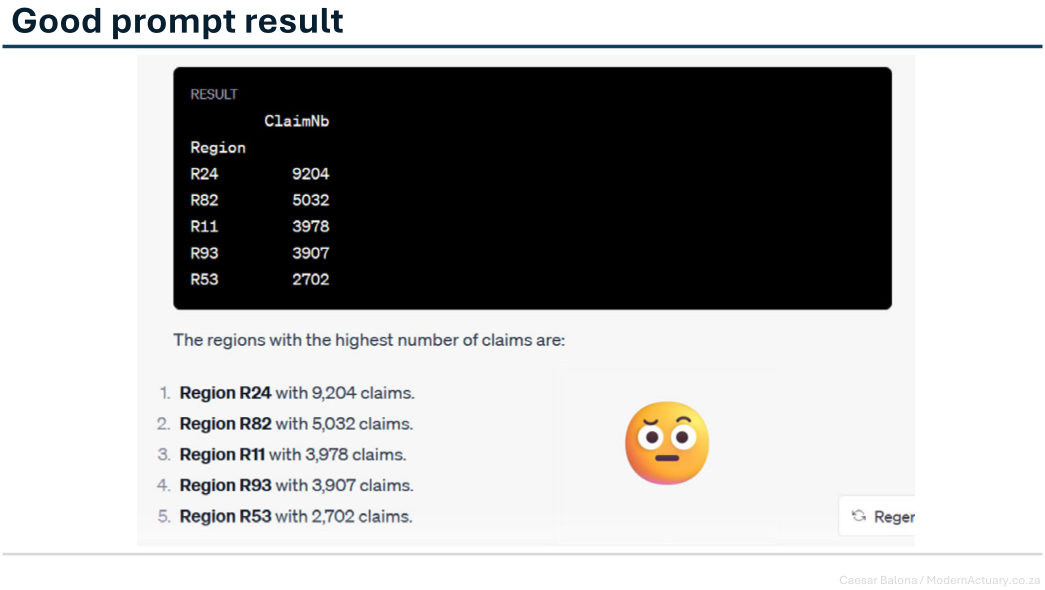
Caesar Balona / ModernActuary.co.za

Figure 3. Prompts.

It is bad because it is vague and short. I did not guide the LLM. The more information and guidance you give, the more benefits you will typically get. A better sample Prompt here would be to give my profession and where I am working. Now, it already knows I am an actuary, that I am working in short-term insurance and my task is to generate a report on claims experience. I have also highlighted some of my concerns.

If I give it this Prompt, I get a much more tailored result (See Figure 4).

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press



Good prompt result

```
RESULT
         ClaimNb
Region
R24         9204
R82         5032
R11         3978
R93         3907
R53         2702
```

The regions with the highest number of claims are:

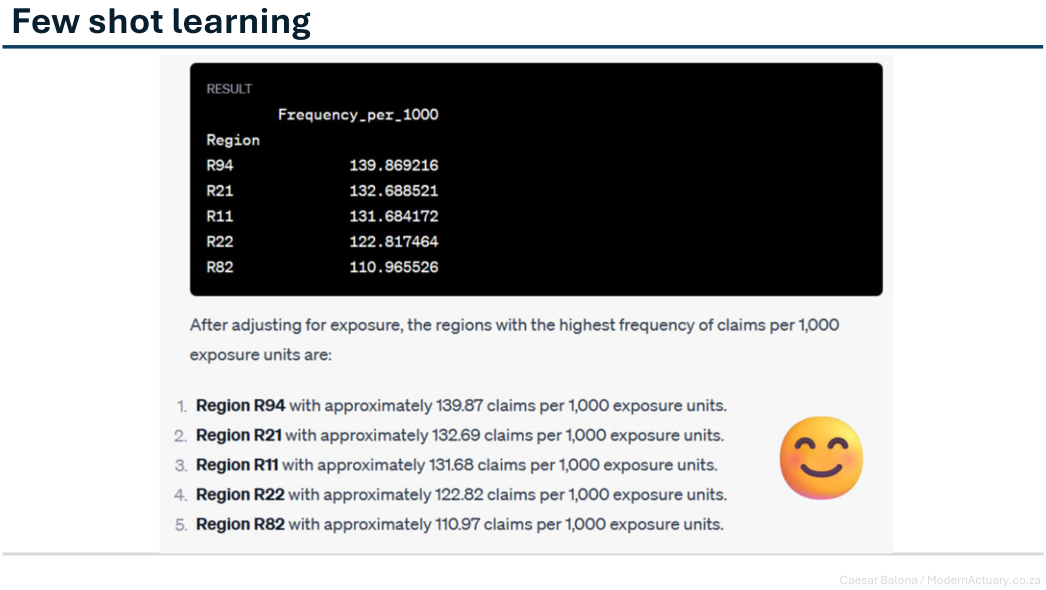
1. **Region R24** with 9,204 claims.
2. **Region R82** with 5,032 claims.
3. **Region R11** with 3,978 claims.
4. **Region R93** with 3,907 claims.
5. **Region R53** with 2,702 claims.

Caesar Balona / ModernActuary.co.za

Figure 4. Good prompt result.

Prompting is one element of interacting with the LLM. Another element is telling it more about the results it is giving you and training it to give you better results. If you are an actuary, you might be thinking 'This is a decent analysis, but there are some things that it is missing.' In that case you, can do something called Few-shot Learning.

Few-shot Learning is telling the model what we want by giving it an example. That could be done by directly giving it the type of results that you are looking for, which can then be applied to other situations. Alternatively, it could be done by giving it appropriate guidance, for example saying 'I want to consider the frequency relative to the exposure. Instead of an absolute number of claims, I want you to consider the frequency of claims per 1,000 exposures.' It can then quickly realise where it has gone wrong and give you a detailed analysis of frequency per 1,000 (See Figure 5).



Few shot learning

```
RESULT
         Frequency_per_1000
Region
R94             139.869216
R21             132.688521
R11             131.684172
R22             122.817464
R82             110.965526
```

After adjusting for exposure, the regions with the highest frequency of claims per 1,000 exposure units are:

1. **Region R94** with approximately 139.87 claims per 1,000 exposure units.
2. **Region R21** with approximately 132.69 claims per 1,000 exposure units.
3. **Region R11** with approximately 131.68 claims per 1,000 exposure units.
4. **Region R22** with approximately 122.82 claims per 1,000 exposure units.
5. **Region R82** with approximately 110.97 claims per 1,000 exposure units.

Caesar Balona / ModernActuary.co.za

Figure 5. Few-shot learning.

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

So, if you are ever using these models and you are not impressed with the output, keep in mind it is important how you prompt the model. It is important for you to prime it and give it examples and feedback on how it is doing. If you do this, you will have a much better experience going forward.

There are other uses for an LLM. You can give it a very complex unstructured problem and ask it to walk you through a solution or help you to brainstorm solutions. You could input a report and ask it to summarise it. It could tutor you or educate you, do data cleaning and preparation, help you develop models, generate documentation, generate stresses and own risk and solvency assessment (ORSA) scenarios, monitor regulation changes, knowledge management . . . . the list goes on.

We have learned that LLMs are tuned to language generation. They are specifically designed for the task of understanding and interpreting language. Websites like ChatGPT allow you easy access to these models. It can be used in various support roles like the coding assistant. But you may be thinking 'How do we directly use this in our work and embed it in our business?' Let us think about that now.

The key to this is something called an API, which stands for Application Programming Interface. It is an interface through which different machines can communicate. This is the key to being able to automate your interaction with an LLM, so that the actuary does not have to type everything into the website to get output. You can embed the LLM into a process and get the results directly.

You could think of the API as being a lot like a Post Office. You send an instruction to that API and it facilitates where that message is sent and how that message is interpreted and given to the system. The system, in this case, is the LLM. The API will take the instruction and send it to the LLM. Then it will take the results from the LLM and send them back to you. It is a layer that facilitates your interaction with the LLM. You will see code examples of this in the paper and in this presentation. The API unlocks a programmatic interface with an LLM, meaning you can embed the LLM within a process.

For example, perhaps a claim arrives in a different language. You can send the claim description to a model that is trained to translate languages. It will translate it and give you the output in your language of choice. Another example is fetching news articles on a regular basis. If you want to track a story over time, it can fetch news articles and summarise them. Another example is extracting detail from documents.

Let us look at some practical examples. In the paper, I go through four case studies. The case studies I will touch on now are identifying emerging risks and parsing reinsurance documents. If you are interested in the other two, they are in the paper. All the code is freely available at: https://github.com/cbalona/actuarygpt-code. You can try it out for yourself.

Let us look at identifying emerging risks. The task is that we want to search for news articles related to, in this case, cyber risk. We want the model to summarise this information for the Board. Once it is summarised, we then want to extract action points and create a project plan for each action point. You can chain together multiple LLMs to do long-chain complex tasks. All four of these steps are performed automatically through using those APIs.

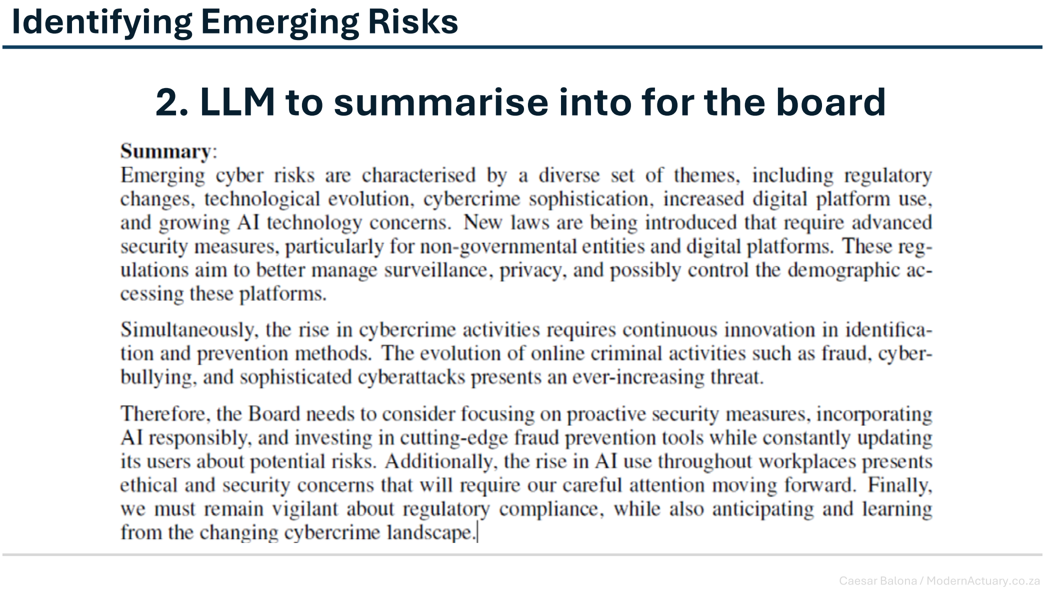
Let us look at the first step – searching for news articles related to cyber risk. This step does not use an LLM at all. It uses a custom Google search engine to find news information related to cyber risk, security risk, and so forth, over the last three months. It sends back a list of links that it finds and a short summary of each of those links from Google.

Next, we want to summarise the information. For that, we write code and give a prompt to the LLM. The prompt is 'I am a risk analyst for a large insurance company. I am tasked with identifying cyber risks. I have collected snippets of these articles, and I want you to identify emerging cyber risks, themes, trends, and so forth. Do not name any companies or individuals for the sake of the presentation and the paper, I do not want to get in any trouble.'

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

I will give it a little bit more information about what I want. I want it to produce a summary and identify the risks. It should be of sufficient length and detail so that a board member can understand the risks and opportunities. I am further telling it that the output will be included in a report to the Board of Directors. This is all in Python code, and I send it through to the LLM.

An example of one of the summaries I got is shown in Figure 6.



Identifying Emerging Risks

2. LLM to summarise into for the board

Summary:
Emerging cyber risks are characterised by a diverse set of themes, including regulatory changes, technological evolution, cybercrime sophistication, increased digital platform use, and growing AI technology concerns. New laws are being introduced that require advanced security measures, particularly for non-governmental entities and digital platforms. These regulations aim to better manage surveillance, privacy, and possibly control the demographic accessing these platforms.

Simultaneously, the rise in cybercrime activities requires continuous innovation in identification and prevention methods. The evolution of online criminal activities such as fraud, cyberbullying, and sophisticated cyberattacks presents an ever-increasing threat.

Therefore, the Board needs to consider focusing on proactive security measures, incorporating AI responsibly, and investing in cutting-edge fraud prevention tools while constantly updating its users about potential risks. Additionally, the rise in AI use throughout workplaces presents ethical and security concerns that will require our careful attention moving forward. Finally, we must remain vigilant about regulatory compliance, while also anticipating and learning from the changing cybercrime landscape.

Caesar Balona / ModernActuary.co.za

Figure 6. Example of LLM summary.

You can see that it gives a nice breakdown and tells us what emerging cyber risks are, how they are characterised, and talks about some laws being introduced, security measures, and so on. It also says that the rise in cybercrime activities requires innovation in identification and prevention methods. Finally, it talks directly to the Board to say it needs to consider focusing on these proactive security measures, on incorporating AI responsibly and on investing in fraud prevention tools, and so forth. Depending on how you frame the prompt, how the summary looks can be quite different and tailored to what you need.

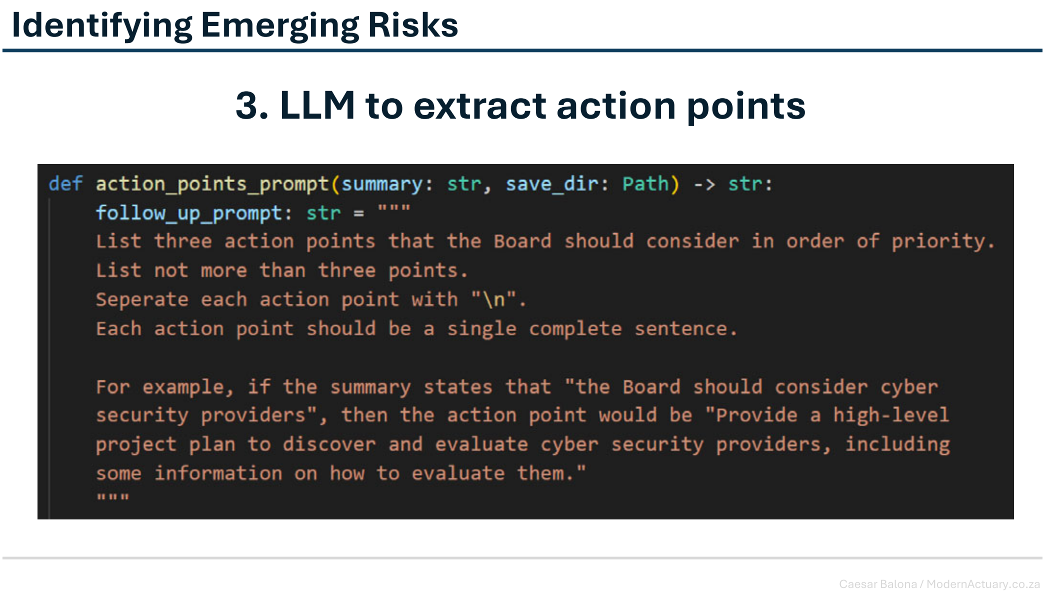
From that summary, we pass the output of one LLM into another LLM, and we say ‘Give us three action points that the board should consider, in order of priority. Do not list more than three points.’

At the bottom of Figure 7, I give it an example. If it states that the Board should consider cybersecurity providers, then I want the action point to provide a high-level project plan to discover and evaluate such providers, and so on.

From this simple example, you can see how the approach can apply in more complex situations. Having that summary that you looked at before, let us now see the action points that the LLM produced. We ask for three action points in a single sentence: ‘We need to formulate a strategic plan to monitor cybercrime, we need to implement a robust ethical framework, and we need to assess existing regulations.’ Finally, we can say ‘Now give us a project plan for each of these points.’

In Figure 8, you can see it is broken down quite well into a project overview and project scope, even breaking it down into different phases. You can then tailor it to your needs.

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press



Identifying Emerging Risks

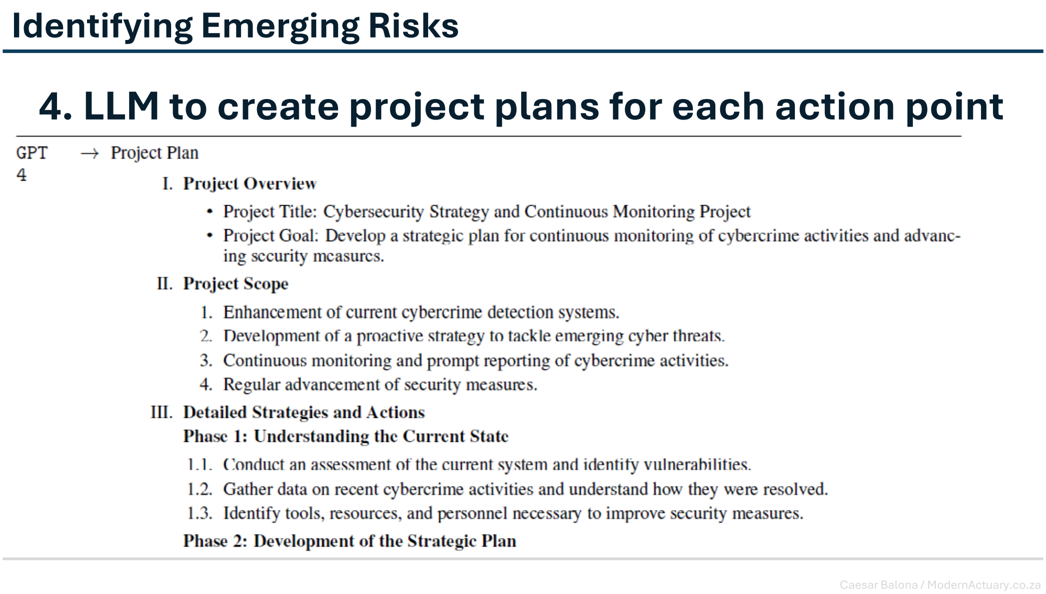
3. LLM to extract action points

```
def action_points_prompt(summary: str, save_dir: Path) -> str:
    follow_up_prompt: str = """
    List three action points that the Board should consider in order of priority.
    List not more than three points.
    Seperate each action point with "\n".
    Each action point should be a single complete sentence.

    For example, if the summary states that "the Board should consider cyber
    security providers", then the action point would be "Provide a high-level
    project plan to discover and evaluate cyber security providers, including
    some information on how to evaluate them."
    """
```

Caesar Balona / ModernActuary.co.za

Figure 7. LLM to extract action points.



Identifying Emerging Risks

4. LLM to create project plans for each action point

GPT 4 → Project Plan

I. **Project Overview**

- Project Title: Cybersecurity Strategy and Continuous Monitoring Project
- Project Goal: Develop a strategic plan for continuous monitoring of cybercrime activities and advancing security measures.

II. **Project Scope**

1. Enhancement of current cybercrime detection systems.
2. Development of a proactive strategy to tackle emerging cyber threats.
3. Continuous monitoring and prompt reporting of cybercrime activities.
4. Regular advancement of security measures.

III. **Detailed Strategies and Actions**

Phase 1: Understanding the Current State

1.1. Conduct an assessment of the current system and identify vulnerabilities.
1.2. Gather data on recent cybercrime activities and understand how they were resolved.
1.3. Identify tools, resources, and personnel necessary to improve security measures.

Phase 2: Development of the Strategic Plan

Caesar Balona / ModernActuary.co.za

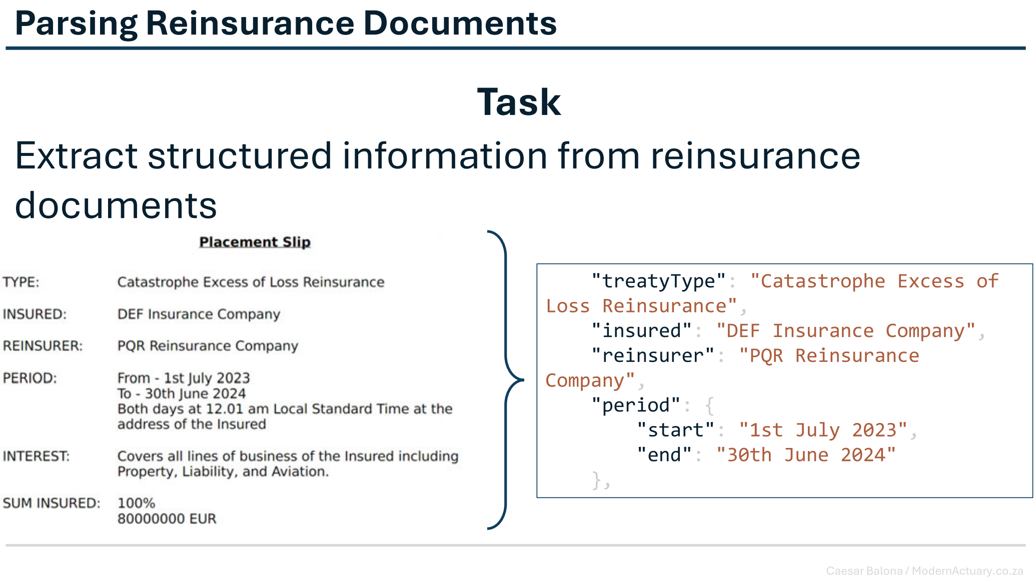
Figure 8. Identifying emerging risks.

I will pause here to summarise what we have done in this case study. We have searched for news articles and summarised them. We have listed action points from that summary, and we have created project plans for each of those points. So, four separate tasks have been completed in one entirely automated, continuous process, using LLM code. One click of the button and all of this happens and you get the results that I have shown you.

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

You can follow the same framework for other applications. For example, I could track the sentiment of my company and see how customers are talking about it. I could search Twitter and the news and get a weekly report of what the sentiment is looking like automatically, over time.

Another example is parsing reinsurance documents or any document for that matter. Here, we want to extract structured information. Some of you will be familiar with reinsurance documents. They can vary in content and length. If you want to understand all the different terms, it can be a laborious task.



Parsing Reinsurance Documents

Task

Extract structured information from reinsurance documents

Placement Slip

TYPE:	Catastrophe Excess of Loss Reinsurance
INSURED:	DEF Insurance Company
REINSURER:	PQR Reinsurance Company
PERIOD:	From - 1st July 2023 To - 30th June 2024 Both days at 12.01 am Local Standard Time at the address of the Insured
INTEREST:	Covers all lines of business of the Insured including Property, Liability, and Aviation.
SUM INSURED:	100% 80000000 EUR

```
    "treatyType": "Catastrophe Excess of
Loss Reinsurance",
    "insured": "DEF Insurance Company",
    "reinsurer": "PQR Reinsurance
Company",
    "period": {
        "start": "1st July 2023",
        "end": "30th June 2024"
    },
```

Caesar Balona / ModernActuary.co.za

Figure 9. Parsing reinsurance documents.

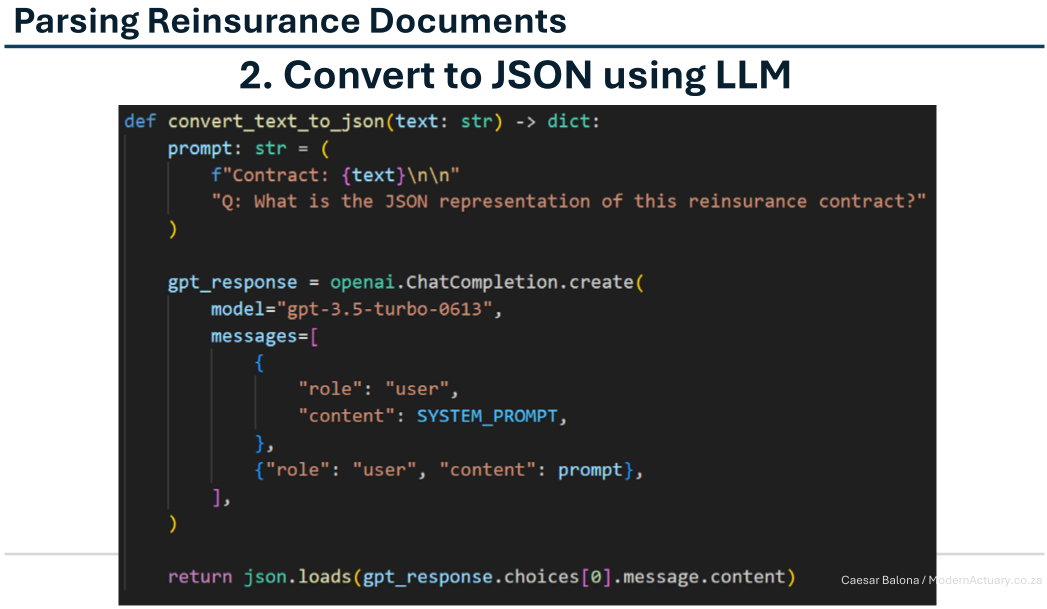
It would be great if we could take a reinsurance slip and extract some well-structured output, like that on the right-hand side of Figure 9. That is an example of JSON output, which is a structured format in which to store information. You can see that we get the treaty type, the insured, the reinsurer, the start and end period, and so forth. So, using LLMs, we input a document and get the structured information. We start with some code that pulls the raw text from the PDF. We take that and input it into an LLM (See Figure 10).

I ask the question, 'What is the JSON representation of this reinsurance contract?' I prime the LLM with a detailed schema of what that JSON structure will look like. I have said 'I want to see the insurer. I want to see the re-insurer. I want to see the start and end period.' I have not given it the information, but I have told it 'These are the things I am looking for. You must fill in the detail.' I give that prompt to the LLM, GPT 3.5, and get a structured JSON output. I can get all the loss layers, the limits, the reinsured percentages, excesses, sum insured, exclusions, and arbitration clauses.

That was an example for reinsurance documents, but could apply to any document from which you need detailed structured information that you can then put into a database or Excel workbook and continue working on. A lot of manual work is being performed quite efficiently here.

Let us pause for another recap. I have shown that we can embed these LLMs into processes using API access. We can perform tasks based on input and output. We can even extract structured output, as in the reinsurance example I have just shown.

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press



Parsing Reinsurance Documents

2. Convert to JSON using LLM

```python
def convert_text_to_json(text: str) -> dict:
    prompt: str = (
        f"Contract: {text}\n\n"
        "Q: What is the JSON representation of this reinsurance contract?"
    )

    gpt_response = openai.ChatCompletion.create(
        model="gpt-3.5-turbo-0613",
        messages=[
            {
                "role": "user",
                "content": SYSTEM_PROMPT,
            },
            {"role": "user", "content": prompt},
        ],
    )

    return json.loads(gpt_response.choices[0].message.content)
```

Caesar Balona / ModernActuary.co.za

Figure 10. Convert to JSON using LLM.

From an actuarial viewpoint, there are risks we need to think about. LLMs are fantastic and extremely powerful but one of the main risks is inherent flaws and biases. Keep in mind that these models are trained on mass amounts of data from the Internet. Whatever flaws and biases you find on the Internet can be replicated in these models. They can produce errors. They can reflect biases in the training data. Critical evaluation of the output is needed for sensitive applications. This is one of the biggest areas of research now in using these models. How do we put protection in place so that these models do not perpetuate biases or give dangerous or erroneous information? Another risk is memory and context. There is only so much information you can put into these models before they get to a point where they are no longer effective. What that means is, in most LLMs, you have a limited context link. The context is, as it sounds, the amount of information it can keep in its mind at a time. That is normally quite small. Currently, it is between 10 and 15 pages for good performance, which sounds like a lot, but when you are going through a lot of information it can quickly be used up. This can result in the model hallucinating and generating nonsense, because it has reached a point where it can no longer understand the context and it is no longer useful. I will touch on how these limits are developing near the end of the presentation.

Another significant risk is data security and privacy. One needs to be very careful of what model one is using in that respect. The catch with the ChatGPT website is that all the information you are putting into it is being used by the company to train their models. You can no longer consider any of the information private. You have given it to them for them to use. This has resulted in a lot of companies banning the use of ChatGPT by their employees. I think that is a good choice because there are safer methods to use.

Finally, ongoing monitoring and validation are needed continuously, especially in this time, where the model is developing very rapidly. You need to review the outputs, the providers themselves, and their associated policies. You may start at a point where you are happy with their policies, data privacy, and the quality of output. But in a short space of time, things can change so drastically that you can no longer trust the output. Validation is essential for these models. They are very powerful so we need to be careful of how we use the output. This all links to professionalism. As actuaries, we have a core responsibility and that is to ensure accuracy and

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

reliability, and act in the best interests of policyholders. This raises some questions and complexities.

How do we ensure that, when using these models, we still comply with our professional standards? How do we manage the risks and avoid unfair discrimination or adverse outcomes? We do not want these models to make decisions or do things that we would not do ourselves in our professional capacity. It is very important that we, as actuaries, enforce our professional code in how we use these models. Then, there is the reality that things can go wrong. My belief is that actuaries bear the onus for the outputs of LLMs, just as we bear the onus for an Excel model that we develop and use to make business decisions. We need to be happy that we can accept that responsibility and mitigate the risks that it brings.

Finally, there is the question of the impact on actuaries. Is it going to take our jobs? My view is that, as actuaries, we are going to have to evolve our skill set. We must understand, most importantly, the limitations, biases, and ethical implications of using these models. We have to adapt to AI technology and take into account these risks because the models will become more and more commonplace, whether we adopt them or not. We have to keep pace and make sure we mitigate the risks they present. That does not mean we need to understand completely how these models work or be able to build them. We have to work collaboratively, with different professions, for example with data scientists and ethicists and engineers, to understand how to effectively use these models. Through all that we need to have a large focus on data privacy, algorithmic fairness, and transparency. We cannot blindly use these models that feature the risks identified without being comfortable with how to mitigate them. It is going to significantly change our workflows and communication. A lot of the time-consuming tasks we do will, most likely, be automated through the use of these models. They have reached a point where tasks that you could not, traditionally, automate with normal programming, you can now effectively automate with an LLM.

That means, as actuaries, we will have increased emphasis on complex reasoning. We are no longer going to be deeply involved in, for instance, data cleaning and manual work. We are going to be thinking about the impacts of these models and what they mean. That brings with it creative problem-solving. Importantly, how do we effectively communicate the results of the LLMs to our stakeholders?

Now, I will discuss some developments since writing the paper. This area is developing rapidly. It is impossible to fully keep pace, but I have picked up the most significant developments and tried to paint a picture of what they mean, particularly with regard to the risks of using this technology.

Artificial General Intelligence (AGI) is an AI that could learn to accomplish any intellectual task that human beings or animals can perform. To put this in comparison with an LLM, an LLM knows the state of the world at a point in time. It knows how to regenerate that information in a structure that suits your task. It does not know how to reason and learn new information to formulate a solution to a problem.

If you are given a completely novel problem, you are able to use different areas of your knowledge to formulate a solution that is completely unique. You can look at different information. For example, you may read a textbook and learn how to use that to craft your solution. LLMs cannot do that. They regurgitate information, but they do it in an exceptionally good way. AGI is the holy grail of AI. That is a model that you can give any problem and it can go out and teach itself how to solve that problem. So, this would be a massive leap forward.

There are many different theories of how close we are to this. I believe we are a lot further forward than most people believe. If we ever reach a point where AGI is achieved, it will have such a massive impact on the world that it will completely change how we approach problem-solving. I thought I might discuss what the future might look like in terms of AI.

There is one approach that is the next step in the direction towards an AGI, and that is the evolution of LLMs. The ‘L’ in LLM stands for ‘Language’. The next step is to expand beyond

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

language and think about audio and video and create what is called a Multi-Modal Model. If LLMs are state-of-the-art now, Multi-Modal Models are what is next. They are trying to achieve AGI by mimicking to a greater extent how humans learn. If you think about LLM, they are just using text. One of the ways humans learn is by reading books. But if you think of a toddler, they learn in so many more ways, such as through visual cues, hearing, touching and feeling, and people talking to them. If we can develop a model that takes in video, audio, and anything else that we can digitise, that is what would constitute a Multi-Modal Model.

Google DeepMind recently released a multi-modal model called Gemini. It has shown very powerful capabilities, but what is most interesting to me is that it does not actually beat the state-of-the-art GPT4 just yet. GPT4 is still the leader in this space, but Gemini is the first view of a multi-modal model.

An important question is whether we trust corporations to hold on to these models and have all of this power? That brings us to the next development, and that is the battle between open source and closed source. Google, OpenAI, Facebook and Apple are closed-source companies that are creating models that you can purchase. But there are companies like Mistral that develop these models and share them openly. They are open-source models. Like the case studies I demonstrated, you can get the code and adapt them. You do not actually have to go through an API. You can have the actual model. I think the next big step in what is happening with LLMs is the battle between open source and closed source. I sit on the open-source side. I think things should be free so that we can build upon them. I think it brings a lot of development to the world.

Finally, Sam Altman, who is the CEO of OpenAI, recently gave an interview and one of the questions was about the risks of LLMs. He said he is not worried about killer robots walking around. He is worried about the very small societal misalignments that may happen without any ill-intention through using these models. These models are so powerful, and they will become so commonplace, that slowly they can change things in society if we do not carefully monitor how we use them. I think that is a key point for actuaries. It relates to the professionalism that I alluded to earlier. Even ChatGPT itself agrees. I asked it ‘Will AI replace actuaries?’ and it said ‘Not entirely.’ The key message is that our skill set goes beyond just crunching numbers. It is about interpreting them. Importantly, who else will double check the AI’s homework? I think that is key to our future in an AI world. We must be custodians and uphold professionalism to make sure that the decisions we make are based on our professional responsibility of protecting policyholders and our financial system.

With that, I will be happy to answer questions. Thank you.

Moderator: Was the code shown in the presentation the original Python?

Mr Balona: Yes, it was all Python code.

Moderator: Do you have any tips for working around a memory and context limitation, in the case of extracting structured information from a reassurance contract that might be more than 50 pages long?

Mr Balona: Currently, we are limited by context. One approach is to build a knowledge base, like case study 3 in the paper. The basic idea is to take your documents and break them into chunks. Then, feed them through a process called embedding, which converts them into a numeric representation that the LLM can search. You then have a database of all of your knowledge. You take your reinsurance document and you add it to this database in a form that the LLM can understand. Then, you prompt the LLM that has been linked to this database. You give it a question and the model will search for the piece of the reinsurance document that is most closely related to your question. It will then load that specific piece into a smaller context and then answer your question. So, that is one approach to solving the context issue.

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

If you want to see an example, refer to case study 3 in the paper, the regulatory knowledge base. I do not give the actual code because I link to a third-party library, so you can get the code from that library.

That is where we were up until late 2023. One of the big focus areas for research in LLMs at the moment is expanding context length. A version 1.5 of the Gemini model was very recently released. It can use up to a million tokens in context, which translates to 750,000 words, which is a lot of information. You can go onto YouTube and search Gemini 1.5 and see all the different examples they give. The next point is where you have models that can take in so much information that you do not have to use a knowledge base. To answer your question, I point to those two areas: using a knowledge base and waiting for models that have a bigger context length.

Moderator: I am interested in your thoughts on developing and training GPTs for specific tasks. I believe this could mitigate some risks and greatly enhance efficiency, while it does not require much coding knowledge.

Mr Balona: I think what is being alluded to by this question is fine-tuning of the models. We have to break this down into different levels. When we fine-tune models, we take an existing model and train it further on specific information and on the specific area that you want to work in. You can do that through OpenAI. Although it does not require a lot of coding knowledge, it is a more technical area and one that is not fully developed yet for the state-of-the-art models. I agree this approach can help to mitigate a lot of the risks. Another option, which is not recommended, is building your own LLM. This is not feasible because of the huge costs involved. You could take an open-source model that is much smaller and tailor it very specifically to a task and help to mitigate the risks. Overall, I agree that fine-tuning is one way to mitigate the risks. In some cases, it does not require coding knowledge, but in other cases it can be quite technical.

Moderator: Are the results generated from an LLM model stable, or do they materially change with different tries?

Mr Balona: Great question. Because they are in principle generative models, they generate information randomly. But if you go through the API approach, most of the models have a few parameters that you can tweak that control the creativity of the model. One parameter is called temperature. The higher the temperature, the more the model is allowed to freely recreate, or freely be creative and create new things. One way to control stability is to reduce the temperature. You will get more stable output, but it will not be as creative, so that might impact problem-solving. Depending on how you parametrise your model, it can be very stable, or it can be not stable at all. If you use ChatGPT, just put in the same prompt a few times and you will see it changes quite a bit. If you want stability, use an API approach with very carefully tuned parameters.

Moderator: You mentioned the future role of actuaries as being custodians of our AI models. Should actuaries have a wider role in the future as custodians of AI models more generally, as opposed to just for those models for which they are currently responsible?

Mr Balona: I would agree. The onus is on the actuary to understand any model in the business processes. The key point is that, because we are accessing a model that is hosted somewhere else, and brought in-house, that does not mean we relinquish the responsibility of having governance around the model. We should be custodians anywhere we make business decisions.

Moderator: In developing embedded processes, what frameworks could you use to evaluate the performance of such processes, for example when evaluating competing prompts, process structures, or model parameters?

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

Mr Balona: This is a broad area. When you approach a problem for which you want to use an LLM, do not go in blindly. I would build up some data first before using the model. For example, say you wanted to extract information from reinsurance documents. First, manually do it for 100 documents, or find a data set where you can see that it has been done. That gives you a training base, and then you can test the performance of your model on that training base. You can use different parameters, run it on the manually extracted information, and see how it performs. That helps with parameters and structures. You can evaluate the prompts by using your training data and see how it performs. Another interesting use case is just using another LLM. One of the best ways to form a prompt is to create an LLM or create a prompt that asks an LLM to evaluate your prompt. You pop in what you want to do, and then you tell it 'Craft a prompt for me that will meet a certain outcome.' That takes away a lot of the work in creating the prompt yourself. It will craft something more tailored that the model will use. Use LLMs that are constrained specifically to those tasks to assist you with that.

Moderator: What are your views on AI/LLMs being used to generate better or more individualised risk data, for example mortality data?

Mr Balona: I think there are two ways to look at that question. One is directly generating the data and the other is using data to generate insights on the individual. In terms of directly generating the data, it is not something I am too confident LLMs would perform very well, because they are more tailored to dealing with general language rather than data. I would say there are better approaches to that. If the question is more pointed at generating data from information on individuals, the devil is in the detail. It can be useful in inferring some information. For example, summarising medical reports or similar materials. But going back to the evaluation part of ensuring that the results you get are high quality, you could get to the point where you are generating information that is pulling you further away from the solution you want and reducing the power of your models.

Moderator: Can these models be adapted to interrogate an Excel-based model? Do you have any examples of use cases from life insurance or pensions, as most of the examples that I have seen are from GI?

Mr Balona: Answering the second question first, that is my fault. I am a GI actuary, so I do not have more examples on the life side. Off the top of my head, I cannot think of any. In terms of interpreting the Excel models, that is a very interesting question. The key there is how you would structure the model so that an LLM could understand it. If you go to my website, there was an article I wrote where I passed in a reserving triangle, which is primarily numerical data, but I formed it in a sentence structure so that the model could now treat it as generating text for a sentence. I do not think it does particularly well, but it is an example of how you can manipulate information into what looks like a sentence to the model. If you can somehow manipulate an Excel model in such a way, whatever that looks like, that the LLM can understand all the different dimensions of it (like the different sheets, the structure and so forth) then I think it could be a powerful tool for understanding spreadsheets. But I think there you might run into context limitations and hallucination as well. It is a great example, and unexplored in my view.

Moderator: Considering the huge focus on privacy matters in the AI space, what are your thoughts when it comes to the AI tools development approach. How do we get started? Industry norm points to an enterprise-level framework, with front end à la GPT, as opposed to actuaries using Python code to leverage the API.

Mr Balona: I think the gold standard of using these models is using the enterprise level that is provided by, for example, Azure from Microsoft. You can get access to the GPT models through Azure. If you trust Microsoft's privacy policies, then you are getting access to enterprise-level

https://doi.org/10.1017/S1357321724000187 Published online by Cambridge University Press

support, and, most importantly, you are getting the infrastructure to run those models in your isolated environment. These models get large and costly to run, so if you have enterprise-level agreement, then I think that is certainly the way to go. It depends on the size of the company. A large company with a big data science team that can implement this at an enterprise level, I think that is the gold standard. If you want to start testing the waters, I think that the API approach with Python is the way to go. With the enterprise level, you will still have access to the Python approach. I think that is key to embedding it in processes. But if you want to just try out the API access itself, most companies that provide it do have strict privacy policies as well. If you are comfortable with the privacy policy they provide, then that is another approach for smaller companies that are trying to dip their feet in.

Moderator: Do you know what regulations/governance is being planned around the use of, and the development of AI, within insurance?

Mr Balona: I do not know in detail. There is a lot of talk, and a lot of investigation, particularly in the US. One of the interesting things is OpenAI, the company that makes GPT, which is the state-of-the-art model, is one of the big companies pushing for regulation. Whether that means that they believe there is a risk, or it means they want to tailor regulation to their needs, is unknown. It is still a very open area but there is no real regulation or governance at this point.

Moderator: Could there be a leap in AI capabilities when/if quantum computing becomes a factor?

Mr Balona: I am not an expert in how quantum computing will impact the training of models. I would presume that if you can train models much more efficiently using quantum computing then you would see a rapid advancement in the development of models. Whether that translates into an actual leap in performance is difficult to say. But certainly, there will be a leap in the pace of development and innovation.

Moderator: Do the LLMs automatically reference online material? You mentioned an example of monitoring regulatory changes. The model would not be able to do this based on its training data. Is the training data fixed and hence becomes out of date?

Mr Balona: Depending on the model you are using, it might have access to the internet or it might not. In most cases, it has a fixed set of data, so it does not have access to the internet. But you can access the internet yourself. If you go to the case study where I searched for news articles, I separately searched the internet and brought that information in a prompt into the LLM. That is how you would get around that situation. Most models do not access the internet live, but you can bring that information in through a separate process.

Moderator: Thanks to Caesar (Balona) and everyone who participated.

References

Balona, C. (2023). ActuaryGPT: Applications of large language models to insurance and actuarial work. Available at SSRN 4543652.

Cite this article: ActuaryGPT: applying LLMs to insurance and actuarial work. *British Actuarial Journal*. https://doi.org/10.1017/S1357321724000187