toot.ai | an explainable AI assistant

JIM EKANEM, DANIELE DI GRANDI, FILIPPO LIBARDI, and OTTO MÄTTAS, Utrecht University, The Netherlands

CCS Concepts: • **Computing methodologies** → **Ontology engineering**; **Reasoning about belief and knowledge**; Machine learning; • **Human-centered computing** → *Human computer interaction (HCI).*

Additional Key Words and Phrases: data sets, neural networks, knowledge representation, text tagging

ACM Reference Format:
Jim Ekanem, Daniele Di Grandi, Filippo Libardi, and Otto Mättas. 2021. toot.ai | an explainable AI assistant. 1, 1 (March 2021), 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 WHAT'S THE PROBLEM?

We are not able to use our time effectively. Most prominently, this can be seen at the workplace. There is an unmet need for technical solutions which can eliminate repetitive and time-consuming mundane tasks for us.

There are many pitfalls for losing time. Avoiding them is often out of the reach of the individual. The work dictates the tasks and the effort required of us. We need to provide services and build products as we have in the past. We need to make sure our revenue stream keeps flowing. But as the business grows, it also gains complexity in deeply-rooted issues which need manual intervention. We spend increasingly more time on solving issues and fixing errors.

As a result, we try to automate issue handling. Unfortunately, this initiative is failing due to the nature of the action of automating itself. It is seen as innovation which rarely brings quick and short-term business value. So, automation is considered as a secondary objective by many if not most business leaders. "Keep it running as long as we can market it to new customers" is a principle familiar to us all.

As we progress, we are creating more recurring and mundane tasks for our bright minds, adding to employee churn and unhappiness, throwing our most valued resource to the time-sinks of those tasks.

Even if there are individuals making an effort to improve processes, the pace of business does not allow for much change anymore. There are multitudes of reasons for this. A very long backlog of more important tasks and the learning curve of adopting new technologies to name a few. Also, pressured managers and business leaders are not making it easier. Innovation is often considered as a by-product and not the driving force of a business. Even if it might have been at some point in the past.

We have only recently started to recognise the value lost due to divided focus and reactive nature of our task-handling. Individuals are starting to raise heads - we need novel situations and learning opportunities to grow. We do not wish to waste our time running into walls or on a never-stopping treadmill. We wish to feel that our efforts have an impact that reaches farther than the bottom-line. We want to create new value.

There must already be a solution to this, right?

There are plenty of different tools and platforms to help us navigate a myriad of specific domains from time management to customer support automation. On the surface, these solutions give us back freedom to a degree.

Under the surface, we continue creating fragmented approaches to the same underlying principle problem - how to best share knowledge and use it to our advantage. As of 2021, there are no commercially available AI Platforms that would allow its patrons to record, validate and act on a problem-solution pair directly. Of course, there are many solutions simplifying task handling on specific domains. For example, there are high-precision recommender systems and navigation tools, while they all lack the ability to make transitive connections between domains. Most open and potentially useful systems are expert systems which are dedicated for enterprise customers. These, again, come with a steep learning curve and often a very high price.

2 HOW CAN WE SOLVE IT?

We are building an interactive AI Platform-as-a-Service. The patron connects the AI assistant to their technological environment so that it can learn how they solve tasks. Eventually the application system has accumulated a large enough knowledge base to know how to solve recurring tasks. Knowledge is expressed in natural language and is human-readable, making it easy to understand. The assistant is then able to infer a solution action from the knowledge by following logic. In collaboration with the authorising patron, these actions can then be executed. With an increasing amount of knowledge, the assistant will eventually handle more tasks for the patron, making their work efficient and more importantly, effective. The overall job specification can be divided into 3 parts.

Firstly, an explainable AI assistant allows the patron to easily record a solution to a certain problem with their preferences in mind. As a result, a subjective knowledge base is learned and consolidated on the platform over time, task by task, no matter the domain.

Secondly, the assistant gives its patron patron easy and direct access to the learned knowledge. This creates the basis for our Explainable AI (XAI) assistant. The step is to mainly validate information stored, and to be updated if something does not add up.

Authors' address: Jim Ekanem, j.e.d.ekanem@students.uu.nl; Daniele Di Grandi, d.digrandi@students.uu.nl; Filippo Libardi, f.m.libardi@students.uu.nl; Otto Mättas, otto@toot.ai, Utrecht University, P.O. Box 80125, Utrecht, Utrecht, The Netherlands, 3508 TC.

XXXX-XXXX/2021/3-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Finally, the assistant can detect an arising problem, explore the knowledge base for a solution and propose to handle those tasks subjectively, after getting a confirmation from its patron. Essentially, it is a DIY AI kit or a human-readable framework for more effective task handling, if you will.

2.1 UNIQUE INNOVATION

Research on knowledge representation and reasoning has gone through decades of motions. We argue that it is now ready to be actively used in a hybrid system together with machine learning methods. We bring together human-understandable knowledge with machine-powered efficiency. We are aiming at technology push innovation [Brem and Voigt 2009].

As argued by [Quiroga 2017], to be creative and intelligent, we have to go beyond merely remembering. We must assimilate concepts and derive meaning. Reasoning helps intelligence emerge through understanding, classification, contextualising and association.

3 WHO ARE WE SOLVING THIS FOR?

The AI services on the market can be divided into Virtual Personal Assitants (VPA), Virtual Customer Assistants (VCA) and Virtual Employee Assistants (VEA). Our vision is that the AI assistant will be offered to individual consumers as well as businesses. However, due to over 50% of consumer's having a negative perception of AI concerning privacy, human relationships and societal impact, we focus on implementing a VEA for businesses. Gartner predicts that VEAs will be used by 25 percent of digital workers in 2021 [Gartner 2019].

In practice, this need is reaffirmed by our collaboration with an international Managed Service Provider (MSP) that struggles with automating small but specific tasks. Right now, there is no way to automatically handle a server disk space issue as there is context to be considered before taking action, every single time. Human reasoning is still needed by the DevOps engineer in the company. The VEA can consider context only in technological environments where it has access to interfaces that provide information. However, target users of the VEA are not limited to IT engineers of a company.

Connected freelancers also described as 'contract professionals' will benefit from our service as well. They are pressured by their relationship to their clients which Gold and Mustafa refer to as client colonisation[Gold and Mustafa 2013]. The prioritisation of work life over private life results in a need for improved use of time by switching off from work. Having a VEA handle certain tasks will support that need. Thus, the above detailed problems specify IT and telecommunication companies as customers of our service.

The market is thriving as global spending on AI in the technology-sector is projected to grow to USD 97 billion in 2023[IDC 2019]. Companies that adopt AI tend to buy their capabilities rather than building them themselves which leads to three types of AI adopters: starters, skilled and seasoned. The main reason for either of them is to make processes more efficient. With seasoned AI adopters investing more than USD 20 million per year. For starters the main difficulty is integrating new AI systems into their current technological infrastructure [Institute and the Deloitte Center for Technology 2020]. Thus, our product is mainly for technology companies that are skilled and seasoned AI adopters.

To also appeal to starters, our product has to be supported by consultancy companies such as IBM's consultancy division IBM Data Science Elite. They help starter companies with their biggest pain point which is AI integration.

Regarding current products on the market, we look at existing solutions by IBM with regards to the three segments of our platform mentioned in section 2.

Firstly, accessing and building a knowledge base for a specific task is one of IBM Watson's capabilities, a cognitive system that focuses on deep Natural Language Processing (NLP) by assessing as much context as possible [Ferrucci 2012].

Secondly, helping users to access and understand an AI output is the key to Explainable AI (XAI). IBM's cloud park has the capabilities to explain, validate and monitor AI models. Companies that used it, reported increased profits totaling $3 million to $12 million over the course of three years as shown by [Research 2020].

Finally, Watson Discovery is used for human-computer collaboration in fields such as hurricane response using pattern recognition, semantic search and chatbot assistance to create a data informed response plan as explained in [IBM [n.d.]].

However, there is no solution that combines the different technologies into a powerful AI assistant that can solve mundane and reoccurring tasks. The presented insights reveal a gap in VEA assistant application that is yet to be filled by an innovative solution.

4 STORIES

In this section, we will take a look at our product with a multi-actor perspective, by individuating Jobs that have to be done with relative Epic Stories and User Stories.

4.1 Job - Help me to optimise my time.

1 Epic Story: When I have loads of tasks to do, I want to carry them out efficiently and effectively, so that I will not work overtime.

1a) User Story: As a CFO of a company, I want to save money on overtime salary, so that I can have a better budget.

1b) User Story: As a member of a family, I want to finish my work on time, so that I can enjoying more time with my family.

1c) User Story: As a HR person of a company, I want my colleagues to be healthy, so that I have lower employee turnover.

4.2 Job - Help me to automate mundane or repetitive tasks.

2 Epic Story: When I have an important task to do, I want to share part of my routine tasks, so that I can better focus on the important one.

2a) User Story: As a freelancer, I want to have a personal assistant manage my time for me, so that I can focus more on customer requests.

2b) User Story: As an employee of a consultant agency, I want to give more focus to the important customers without overlooking the regular ones, so that I can satisfy all the customers accordingly to their expectations.

3 Epic Story: When there is a repetitive task assigned to me, I want an assistant to recognise the repetition, so that it can handle it for me.

3a) User Story: As an engineer, I do not want to deal with repetitive tasks, so that I can develop new features.

4.3 Job - Help me to improve the quality of my work.

4 Epic Story: When I am not capable of doing a task, I want some help to cover the lacking abilities, so that I can improve my work performances and learn from it.

4a) User Story: As a freelancer, I want to have a personal assistant manage my time for me, so that I can focus more on customer requests.

4b) User Story: As a freelancer, I want to install to my customers the best possible solution even though I don't know which is, so that I can offer the best possible service.

4c) User Story: As an employee of a consultant agency, I want my knowledge to be always up to date, so that I can improve my customers experience.

5 Epic Story: When I am solving a problem, I want to know the best solution, so that the final result will be better.

5a) User Story: As an employee in a non-profit company, I want to solve my problem with the best solution, so that I can make a positive impact to the world.

5 TECHNOLOGY

In this section, we will proceed with an explanation and an overview of how we intend to build our assistant platform, diving in its technological functionalities and architecture.

5.1 FUNCTIONAL FEATURES

Here, we will provide a general overview of how the assistant will work. The details on the technology used in order to build our platform are left for section 5.2. The process we are describing here is graphically summarised in a BPMN diagram (1) for visual reference.

As previously mentioned, the aim of this project is to build an AI assistant that tries to solve tasks previously solved by the patron. Accordingly, the system will record patron's action and try to fit the newly acquired information into an ontology knowledge representation of the given environment. Once the new experience is solidly formalised in a knowledge base, the system will be able to replicate it again. Between the agent and the patron it will sit an interface that will take up the job of not only making the assistant a pleasant presence within the environment, but also it should feature a lightweight framework in order to favor data ingestion speed. Information ingested by the system need to be schematised, more precisely the system needs to extract features from the scenario, learn relationship between these features and update its own internal ontology. In order to achieve this, we are exploring methods and frameworks used in the field of ontology learning. This is a technique that combines Machine Learning with Ontology construction and inferential processes and it really seems to be a feasible way to go [Asim et al. 2018].

After the initial "training" performed by the patron, in order to make the assistant capable of infer decisions based on the knowledge it acquired, there is the validation task: the patron should rate the assistant decisions before they are applied, in a feedback loop that allows the assistant to recalibrate its inner weights and update its knowledge based on the rate got by the patron and the ground truth. The nice thing about this solution is that the knowledge database is stored in cloud, allowing multiple patrons to input as well as extract knowledge, in a system that is in constant evolution and improvement.

5.2 TECHNICAL FEATURES

Nowadays, cloud computing is influencing the IT landscape and becoming a significant economic factor for software producing organisations.

There are many aspects to consider in leveraging cloud technologies. As presented by [Au 2016] the list of benefits includes but is not limited to cost savings, easy implementation, storage capacity, accessibility, reliability and manageability. On the flip side, one has to consider vendor lock-in as the extensive platforms bring vendor-specific tool sets. Additionally, extra care has to be taken with security. Whenever using the internet, security is not guaranteed – even if the cloud service providers can implement the best security standards and industry certifications.

Cloud computing is a fast-growing technology in a non-transparent market with diverse vendors, each of them having their specific services and deployment models as explained by [Farshidi et al. 2018]. Typically, the service portfolios are heterogeneous and combined with complicated service features and pricing models.

Even though the benefits are clear for building a scalable platform, using cloud technologies brings a challenge. Namely, evaluating and selecting the most suitable Infrastructure-as-a-Service Cloud Provider for software producing organisations according to their preferences and requirements. The same applies to us.

As concluded by [Farshidi et al. 2018], finding a feasible solution for the Infrastructure-as-a-Service Cloud Provider selection problem based on decision-makers' priorities and requirements requires deep investigation into the documentation of cloud vendors and extensive expert analysis. As we have extensive experience with Amazon Web Services, further referred to as AWS, we have chosen to leverage the tools provided by this particular Cloud Provider.

Main parts of the solution have been described on figure 2 as modules. Furthermore, figure 3 shows distinct services provided by AWS that we are using for our architecture. Figure 4 is there to help directly navigate the architecture graph by providing a legend. Below, we are following the modules and giving an overview of the architecture.

5.2.1 Patron spacing. Our platform is available through many channels to different actors. Amazon Cognito lets you add patron sign-up, sign-in, and access control to your web and mobile apps quickly

and easily. Amazon Cognito scales to millions of patrons and supports sign-in with social identity providers, such as Apple, Facebook, Google, and Amazon, and enterprise identity providers via SAML 2.0 and OpenID Connect.

Amazon Lex is a service for building conversational interfaces into any application using voice and text. Amazon Lex provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognise the intent of the text, to enable you to build applications with highly engaging patron experiences and lifelike conversational interactions. With Amazon Lex, the same deep learning technologies that power Amazon Alexa are now available to any developer, enabling you to quickly and easily build sophisticated, natural language, conversational bots ("chatbots").

AWS IoT Core lets you connect IoT devices to the AWS cloud without the need to provision or manage servers. AWS IoT Core can support billions of devices and trillions of messages, and can process and route those messages to AWS endpoints and to other devices reliably and securely. With AWS IoT Core, your applications can keep track of and communicate with all your devices, all the time, even when they aren't connected.

AWS IoT Core also makes it easy to use AWS and Amazon services like AWS Lambda, Amazon Kinesis, Amazon S3, Amazon SageMaker, Amazon DynamoDB, Amazon CloudWatch, AWS CloudTrail, Amazon QuickSight, and Alexa Voice Service to build IoT applications that gather, process, analyse and act on data generated by connected devices, without having to manage any infrastructure.

Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. APIs act as the "front door" for applications to access data, business logic, or functionality from your backend services. Using API Gateway, you can create RESTful APIs and WebSocket APIs that enable real-time two-way communication applications. API Gateway supports containerised and serverless workloads, as well as web applications.

API Gateway handles all the tasks involved in accepting and processing up to hundreds of thousands of concurrent API calls, including traffic management, CORS support, authorisation and access control, throttling, monitoring, and API version management. API Gateway has no minimum fees or startup costs. You pay for the API calls you receive and the amount of data transferred out and, with the API Gateway tiered pricing model, you can reduce your cost as your API usage scales.

Amazon Kinesis Data Streams (KDS) is a massively scalable and durable real-time data streaming service. KDS can continuously capture gigabytes of data per second from hundreds of thousands of sources such as website clickstreams, database event streams, financial transactions, social media feeds, IT logs, and location-tracking events. The data collected is available in milliseconds to enable real-time analytics use cases such as real-time dashboards, real-time anomaly detection, dynamic pricing, and more.

5.2.2 Data ingestion. After the patron has been authenticated and the necessary information has been made available for the platform pipeline, we move towards working directly with the data.

AWS Lambda is a serverless compute service that lets you run code without provisioning or managing servers, creating workload-aware cluster scaling logic, maintaining event integrations, or managing runtimes. With Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code as a ZIP file or container image, and Lambda automatically and precisely allocates compute execution power and runs your code based on the incoming request or event, for any scale of traffic. You can set up your code to automatically trigger from 140 AWS services or call it directly from any web or mobile app. You can write Lambda functions in your favorite language (Node.js, Python, Go, Java, and more) and use both serverless and container tools, such as AWS SAM or Docker CLI, to build, test, and deploy your functions. We are using Python to develop our own functions for our own specific needs.

5.2.3 Knowledge extraction. After we have ingested patron data, we move on to extracting knowledge from it.

Amazon Comprehend is a natural language processing (NLP) service that uses machine learning to find insights and relationships in text.

There is a treasure trove of potential sitting in your unstructured data. Patron emails, support tickets, product reviews, social media, even advertising copy represents insights into patron sentiment that can be put to work for your business. The question is how to get at it? As it turns out, Machine learning is particularly good at accurately identifying specific items of interest inside vast swathes of text (such as finding company names in analyst reports), and can learn the sentiment hidden inside language (identifying negative reviews, or positive patron interactions with patron service agents), at almost limitless scale.

Amazon Comprehend uses machine learning to help you uncover the insights and relationships in your unstructured data. The service identifies the language of the text; extracts key phrases, places, people, brands, or events; understands how positive or negative the text is; analyses text using tokenisation and parts of speech; and automatically organises a collection of text files by topic. You can also use AutoML capabilities in Amazon Comprehend to build a custom set of entities or text classification models that are tailored uniquely to your organisation's needs.

Amazon Rekognition makes it easy to add image and video analysis to your applications using proven, highly scalable, deep learning technology that requires no machine learning expertise to use. With Amazon Rekognition, you can identify objects, people, text, scenes, and activities in images and videos, as well as detect any inappropriate content. Amazon Rekognition also provides highly accurate facial analysis and facial search capabilities that you can use to detect, analyse, and compare faces for a wide variety of patron verification, people counting, and public safety use cases.

With Amazon Rekognition Custom Labels, you can identify the objects and scenes in images that are specific to your business needs. For example, you can build a model to classify specific machine parts on your assembly line or to detect unhealthy plants. Amazon Rekognition Custom Labels takes care of the heavy lifting of model

development for you. You simply need to supply images of objects or scenes you want to identify, and the service handles the rest.

5.2.4 Knowledge representation. After having extracted concepts from the data, we move towards creating a knowledge base.

We store the data in a graph database. Amazon Neptune is a fast, reliable, fully managed graph database service that makes it easy to build and run applications that work with highly connected datasets. The core of Amazon Neptune is a purpose-built, high-performance graph database engine optimised for storing billions of relationships and querying the graph with milliseconds latency. Amazon Neptune supports popular graph models Property Graph and W3C's RDF, and their respective query languages Apache TinkerPop Gremlin and SPARQL, allowing you to easily build queries that efficiently navigate highly connected datasets. Neptune powers graph use cases such as recommendation engines, fraud detection, knowledge graphs, drug discovery, and network security.

Amazon Neptune is highly available, with read replicas, point-in-time recovery, continuous backup to Amazon S3, and replication across Availability Zones. Neptune is secure with support for HTTPS encrypted client connections and encryption at rest. Neptune is fully managed, so you no longer need to worry about database management tasks such as hardware provisioning, software patching, setup, configuration, or backups.

Additionally, we will be creating a representation in the W3C Web Ontology Language (OWL). It is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be exploited by computer programs, e.g., to verify the consistency of that knowledge or to make implicit knowledge explicit. OWL documents, known as ontologies, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies. OWL is part of the W3C's Semantic Web technology stack, which includes RDF, RDFS, SPARQL, etc.

5.2.5 Knowledge validation. After having created a knowledge base, we move towards validating the concepts. In this task, the patron is supported in his feedback loop by the following back-end visualisation and searching tools.

Amazon Kendra is an intelligent search service powered by Machine learning. Kendra reimagines enterprise search for your websites and applications so your employees and patrons can easily find the content they are looking for, even when it's scattered across multiple locations and content repositories within your organisation.

Using Amazon Kendra, you can stop searching through troves of unstructured data and discover the right answers to your questions, when you need them. Amazon Kendra is a fully managed service, so there are no servers to provision, and no machine learning models to build, train, or deploy.

For more capabilities, we will use the ELK stack. Amazon Elasticsearch Service is a fully managed service that makes it easy for you to deploy, secure, and run Elasticsearch cost effectively at scale. You can build, monitor, and troubleshoot your applications at the scale you need. The service provides support for open source Elasticsearch APIs, managed Kibana, integration with Logstash and other AWS services, and built-in alerting and SQL querying. Amazon Elasticsearch Service lets you pay only for what you use – there are no upfront costs or usage requirements. With Amazon Elasticsearch Service, you get the ELK stack you need, without the operational overhead.

We are also sharing updates to the knowledge base directly with the patron via a stream. Neptune Streams logs every change to your graph as it happens, in the order that it is made, in a fully managed way. Once you enable Streams, Neptune takes care of availability, backup, security and expiry.

We allow the patrons to validate the knowledge via multiple channels. Amazon Pinpoint is a flexible and scalable outbound and inbound marketing communications service. You can connect with patrons over channels like email, SMS, push, or voice. Amazon Pinpoint is easy to set up, easy to use, and is flexible for all marketing communication scenarios. Segment your campaign audience for the right patron and personalise your messages with the right content. Delivery and campaign metrics in Amazon Pinpoint measure the success of your communications. Amazon Pinpoint can grow with you and scales globally to billions of messages per day across channels.

5.2.6 Reasoning and Knowledge output. After we have validated the knowledge, we move the information to a data lake.

AWS Lake Formation is a service that makes it easy to set up a secure data lake in days. A data lake is a centralised, curated, and secured repository that stores all your data, both in its original form and prepared for analysis. A data lake enables you to break down data silos and combine different types of analytics to gain insights and guide better business decisions.

However, setting up and managing data lakes today involves a lot of manual, complicated, and time-consuming tasks. This work includes loading data from diverse sources, monitoring those data flows, setting up partitions, turning on encryption and managing keys, defining transformation jobs and monitoring their operation, re-organising data into a columnar format, configuring access control settings, deduplicating redundant data, matching linked records, granting access to data sets, and auditing access over time.

Creating a data lake with Lake Formation is as simple as defining data sources and what data access and security policies you want to apply. Lake Formation then helps you collect and catalog data from databases and object storage, move the data into your new Amazon S3 data lake, clean and classify your data using machine learning algorithms, and secure access to your sensitive data. Your patrons can access a centralised data catalog which describes available data sets and their appropriate usage. Your patrons then leverage these data sets with their choice of analytics and machine learning services, like Amazon Redshift, Amazon Athena, and (in beta) Amazon EMR for Apache Spark. Lake Formation builds on the capabilities available in AWS Glue.

From there, we can create off-the-shelf machine learning models for addressing a certain problem domain.

Amazon SageMaker is a machine learning service that you can use to build, train, and deploy ML models for virtually any use case. The ML tool set brings together all the relevant steps for creating ML models - preparing data, building notebooks, training and tuning models, deploying and managing them in end applications. It comes with a vast range of different products embedded in the kit. We are still looking at which tools we need for our platform.

After incorporating data and machine learning models in the data lake, we output a solution to the patron. This step includes reasoning on the ontology and inferring a solution as explained in 5.2.4.

Amazon Personalize enables developers to build applications with the same machine learning (ML) technology used by Amazon.com for real-time personalised recommendations.

Amazon Personalize makes it easy for developers to build applications capable of delivering a wide array of personalisation experiences, including specific product recommendations, personalised product re-ranking, and customised direct marketing. Amazon Personalize is a fully managed machine learning service that goes beyond rigid static rule based recommendation systems and trains, tunes, and deploys custom ML models to deliver highly customised recommendations to patrons across industries such as retail and media and entertainment.

Amazon Personalize provisions the necessary infrastructure and manages the entire ML pipeline, including processing the data, identifying features, using the best algorithms, training, optimising, and hosting the models. You will receive results via an Application Programming Interface (API) and only pay for what you use, with no minimum fees or upfront commitments. All data is encrypted to be private and secure, and is only used to create recommendations for your patrons.

Again, we leverage Amazon Pinpoint to deliver the results.

And finally, we can present the results to the patron via speech.

Amazon Polly is a service that turns text into lifelike speech, allowing you to create applications that talk, and build entirely new categories of speech-enabled products. Polly's Text-to-Speech (TTS) service uses advanced deep learning technologies to synthesise natural sounding human speech. With dozens of lifelike voices across a broad set of languages, you can build speech-enabled applications that work in many different countries.

In addition to Standard TTS voices, Amazon Polly offers Neural Text-to-Speech (NTTS) voices that deliver advanced improvements in speech quality through a new machine learning approach. Polly's Neural TTS technology also supports two speaking styles that allow you to better match the delivery style of the speaker to the application: a Newscaster reading style that is tailored to news narration use cases, and a Conversational speaking style that is ideal for two-way communication like telephony applications.

Finally, Amazon Polly Brand Voice can create a custom voice for your organisation. This is a custom engagement where you will work with the Amazon Polly team to build an NTTS voice for the exclusive use of your organisation.

Furthermore, in order to connect all these modules together (where is needed), we will be using Python as a programming language, so that we can take further advantages by exploiting all the vast world of AI and machine learning packages that Python's patrons have coded in the years.

REFERENCES

Muhammad Nabeel Asim, Muhammad Wasim, Muhammad Usman Ghani Khan, Waqar Mahmood, and Hafiza Mahnoor Abbasi. 2018. A survey of ontology learning techniques and applications. *Database* 2018 (10 2018). https://doi.org/10.1093/database/bay101 arXiv:https://academic.oup.com/database/article-pdf/doi/10.1093/database/bay101/27329264/bay101.pdf bay101.

Regina Au. 2016. To Cloud Compute, or Not to Cloud Compute? *Innovations in Pharmaceutical Technology* (07 2016), 32–35.

Alexander Brem and Kai-Ingo Voigt. 2009. Integration of market pull and technology push in the corporate front end and innovation management—Insights from the German software industry. *Technovation* 29, 5 (2009), 351–367. https://doi.org/10.1016/j.technovation.2008.06.003 Technology Management in the Service Economy.

Siamak Farshidi, Slinger Jansen, Rolf Jong, and Sjaak Brinkkemper. 2018. A Decision Support System for Cloud Service Provider Selection Problem in Software Producing Organizations. https://doi.org/10.1109/CBI.2018.00024

D. A. Ferrucci. 2012. Introduction to "This is Watson". *IBM Journal of Research and Development* 56, 3.4 (2012), 1:1–1:15. https://doi.org/10.1147/JRD.2012.2184356

Gartner. 2019. Gartner predicts 25 percent of digital workers will use virtual employee assistants daily by 2021. *Press releases* (Jan 2019). Retrieved March 2, 2021 from https://www.gartner.com/en/newsroom/press-releases/2019-01-09-gartner-predicts-25-percent-of-digital-workers-will-u

Michael Gold and Mona Mustafa. 2013. 'Work always wins': client colonisation, time management and the anxieties of connected freelancers. *New Technology, Work and Employment* 28, 3 (2013), 197–211. https://doi.org/10.1111/ntwe.12017 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/ntwe.12017

IBM. [n.d.]. *Watson discovery use cases.* Retrieved March 2, 2021 from https://www.ibm.com/cloud/watson-discovery/use-cases

International Data Corporation IDC. 2019. Worldwide Spending on Artificial Intelligence Systems Will Be Nearly $98 Billion in 2023, According to New IDC Spending Guide. *IDC press relations* (sep 2019). Retrieved March 2, 2021 from https://www.idc.com/getdoc.jsp?containerId=prUS45481219

Deloitte AI Institute and Media & Telecommunications the Deloitte Center for Technology. 2020. Thriving in the era of pervasive AI. *Deloitte's state of AI in the Enterprise* 3 (July 2020).

Rodrigo Quian Quiroga. 2017. *The Forgetting Machine: Memory, Perception, and the 'Jennifer Aniston Neuron'.* Benbella Books Inc., Dallas, Texas, U.S.

Forrester Research. 2020. New Technology: The Projected Total Economic Impact Of Explainable AI And Model Monitoring In IBM Cloud Pak For Data. (aug 2020).

Tjerk Spijkman, Sjaak Brinkkemper, Fabiano Dalpiaz, Anne-Fleur Hemmer, and Richard Bospoort. 2019. Specification of Requirements and Software Architecture for the Customisation of Enterprise Software: A Multi-case Study Based on the RE4SA Model. 64–73. https://doi.org/10.1109/REW.2019.00015

A BPMN DIAGRAM

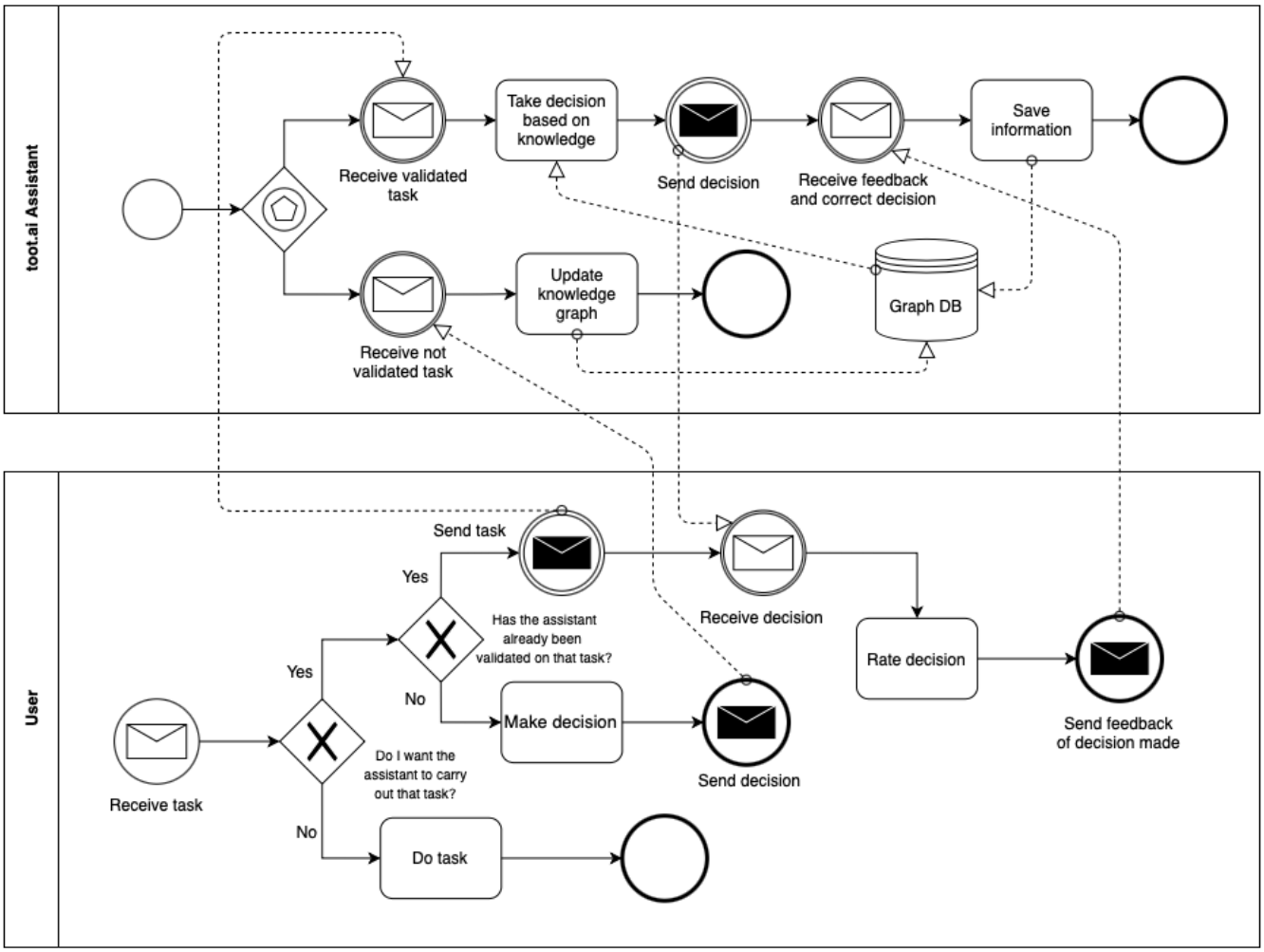

Fig. 1. Graphical representation for specifying the business processes in the business process model.

B PRODUCT MODULES GRAPH

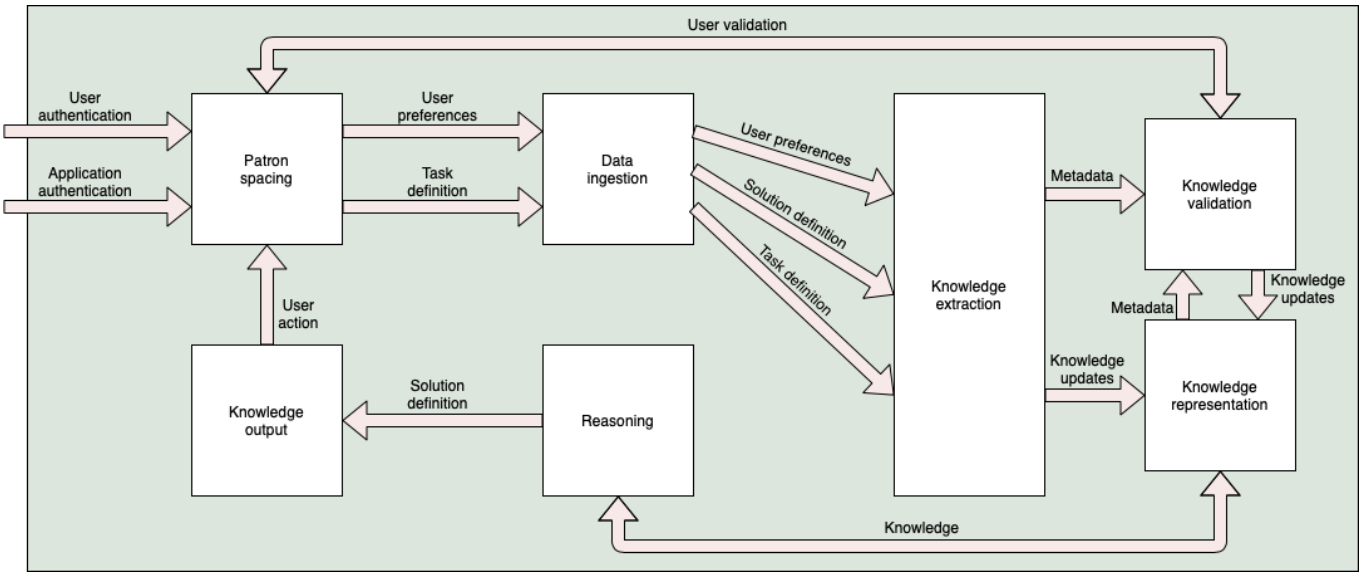

Fig. 2. Graphical representation of the functional architecture through defining software modules as proposed by [Spijkman et al. 2019].

C ARCHITECTURE

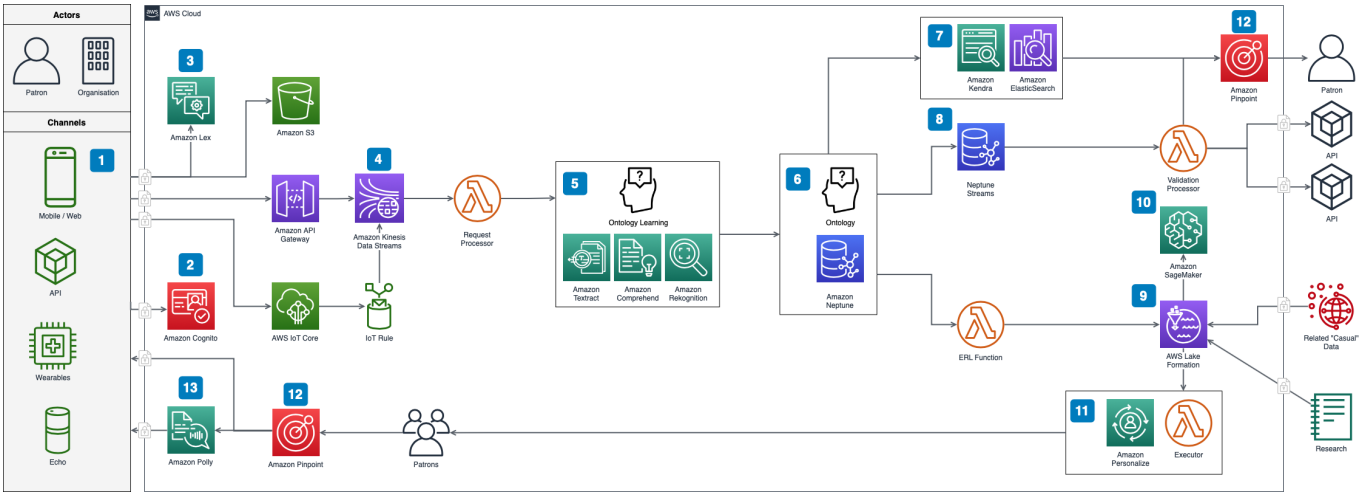

Fig. 3. Graphical representation of the platform architecture built on Amazon Web Services.

1. toot.ai is available through many channels to different actors. Individual patrons approach individual assistants while organisations approach an environment with the possibility for on-prem knowledge.
2. Add patron sign-up, sign-in, and access control to our web and mobile apps quickly and easily. Supports sign-in with social identity providers and enterprise identity providers via SAML 2.0 and OpenID Connect.
3. Automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognise intent of the text.
4. Real-time data streaming service. All patron preferences together with task requests are captured by continuously ingesting data which is delivered through Amazon API Gateway or AWS IoT Core.
5. Ontology Learning. NLP, Physical documents, Video and Image entity extraction and load into a graph.
6. Graph database service. It supports W3C's RDF and SPARQL. OWL is still needed for proper inference functionality.
7. Knowledge direct validation by patrons. Also used for debugging and continuous improvement.
8. Knowledge update validation. Neptune Streams capture changes to our graph (change-log data) as they happen.
9. AWS Lake Formation stores all our data, both in its original form and prepared for analysis.
10. We leverage Amazon SageMaker for learning ML models to have an off -the-shelf solution.
11. We send back the proper solution to the patron with recommendations for the specific user.
12. We leverage an outbound and inbound marketing communications service. We can connect with patrons over channels like email, SMS, push, or voice.
13. Text-to-Speech (TTS) service synthesises natural sounding speech for connecting to the patrons.

Fig. 4. Legend to explain specific components of the platform architecture.