We develop chatbots that can learn, contextualize,
and use data from past conversations to execute actions and
make relevant suggestions.

Kore.ai's Conversational AI powered Chatbots enable Complex Human Conversations

42% of service agents are unable to resolve customer issues due to disconnected systems, archaic user interfaces, and multiple applications- Salesforce.

Conversational Intelligence of Chatbots is a key component of successful dialog and context management.

Bots must be designed to adapt to human conversational styles, including ability to detect multiple intents and sub-intents.

Bots should be able to understand context, follow-up interrupted intents and dynamically amend entities.

"So pretty much everyone today who's building applications, whether they be mobile apps or desktop apps or websites, will build bots as the new interface where you have a human dialogue interface versus menus of the past"

--Microsoft CEO Satya Nadella in 2018

Our Intelligent Chatbots can decipher natural human speech.

It is easy to build a robot that understands simple and clear directions. But that's not how humans speak. In natural conversation, we tend to interrupt ourselves, omit essential information, switch topics and change our minds without warning.

Our chatbots are capable of understanding and effectively handling these challenges.

Interruptions

The Kore.ai platform easily handles interruptions, clarifications and similar nuances of human conversations. Where multiple tasks are involved in a single user utterance, the bot breaks down the intents and sequences the tasks or prompts the users for next-step guidance. Users have granular control over their conversations and bots are designed to respond intelligently.

Intelligent Context Switching (Hold & Resume)

We humans have a habit of continuing a conversation from the point where we last left off. That means that bots need to have the ability to hold the current task, start another task, complete and then resume the original task. Kore.ai provides just that: we ensure conversational continuity across sessions and across channels.

The developers can also choose to enable or disable the setting and customize the conversation by training the bot to either hold & resume a task (or) stop newer tasks from interrupting the current task (or) prompt the end customer to choose the action.

Dialog Continuity

Human brain is wired to understand the implicit logic and context for an action, but for chatbots this is a complex task. Kore chatbots have a granular control over managing the context of a conversation and the ability to bring it to its logical ending.

The developer can define a significant number of linked tasks while designing & training a chatbot, which can be invoked anytime during a conversation when detected.

For example, a customer's enquiry over balance in the account can be linked to an action task such as 'transfer' / 'payment', as that would lead it to a logical ending.
Developer will have an ability to map same target task to a source task multiple times and will need to define different Pre-conditions. The condition that is matched will only be considered.

In the above example:
Source task: Balance enquiry
Target Task: Transfer or Payment

Follow up Intents

Human conversations tend to switch between intents and entities, often combining multiple things into one. Kore.ai enhanced Bot Context Switching abilities provide developers with full control over handling such switches and defining the overall conversational experience. This includes:
(a) ability to add conditional exceptions between tasks while passing contextual data between them.
(b) using Follow-up Intents, enabling the bot to capture all unattended interruptions from a conversation flow and keeping them accessible for the future.

Amend Entity

User conversations are dynamic with multiple intentions in the single utterance.
Kore.ai chatbots have the ability to automatically amend entity value during a conversation.
The bot understands the context, identifies the entity to be amended and either clarifies the action to be taken or takes action directly.

Multiple Intent Detection

It is common for human conversations to have multiple intents. Our platform is the only one that can identify and order, and handle multiple intents at the dialog management layer.

Our Chatbots understand context

Context is critical to human conversations. The bot needs to be aware of the context at various levels such as enterprise, user, session and bot levels. Our bots are designed to retain short term and long-term memory to have intuitive conversations with users.

Redefine Customer Experience with Context

We make smarter chatbots that can leverage relevant data in the right circumstances.
Kore.ai's platform supports multiple context variables to allow for thoughtful design and complete control of how this information gets used in distinct scenarios. Through the Bot Builder tool, developers can customize, categorize and apply contextual information that maximizes the chatbot's ability to make intelligent decisions and avoid confusion or making unnecessary steps.

User context

User Context: Individual user information or preferences that can be shared by all enterprise bots the user will interact with.
Why is this useful? Default user information and preferences can be saved and used in varying tasks, and travel with the user from bot-to-bot, to make transactions and interactions shorter and easier.

Example use case: Information such as a user's home address, payment information, an airline seat, or baggage preference, etc. could be user context stored and remembered for relevant circumstances.

Enterprise context

Enterprise Context: Information that represents company-wide rules and standards that apply to all users and bots.
Why is this useful? Allows developers to ensure bots keep company requirements in context and enforced when needed.
Example use case: A company's travel policy, or expense limits could be applied as enterprise context that overrides an employee's preferences when using a travel bot.

Bot context

Bot Context: User or task information dynamically captured at a bot level that can be used in context with some or all of the users of that bot.
Why is this useful? Allows a developer to use information that the bot collects dynamically about users or tasks – such as user or task counts and transaction values – to design additional logic specific to that bot.
Example use case: A customer-facing shopping bot could collect information about the number of users that purchase a product on a given day, and design logic that says to offer every 100th user a promotion or discount.

Session context

Bot Context: Our virtual agent has an ability to store developer defined user preferences in memory and use them in conversations. The Information can be stored and retrieved from session context.
Why is this useful? Session context allows the user to avoid repeating information that's relevant if they complete multiple tasks in a single session.
Example use case: A bank customer says, "How much do I have in checking?" and then asks to "Transfer $500 to savings." The bot could recognize "checking" as session context and thus make the transfer without asking, "From what account?"

Our Chatbots Can Learn

The Kore.ai Bots Platform facilitates learning as a component of our natural language engine. Since every interaction gets logged and categorized as a success or failure, bots can be designed to learn from those lessons and be adjusted accordingly, either through human intervention or automatically.

Supervised Learning for NL

Supervised learning for NL aids bot intelligence by allowing a human to analyze and respond to the way people communicate with the bot once it's up and running. Through our Bot Builder tool, developers and admins can evaluate all interaction logs, easily change NL settings for failed scenarios, and use these scenarios to retrain the bot for better conversations. Developers can also leverage chat logs to build predictive models and use the outcomes to further define additional proactive alerts, suggested actions, or automated workflows.

Unsupervised Learning for NL

Unsupervised learning for NL can be applied to expand the language capabilities of your chatbot – without human intervention. Unlike other unsupervised models in which chatbots learn from any input – good or bad – the Kore.ai Bots Platform enables chatbots to automatically increase vocabulary only when the chatbot successfully recognizes the intent and extracts the entities of a human's request to complete a task.