

# IBM Cloud の Watson Assistant を利用した会話型情報提供について

佐野寿久

情報通信技術支援室 情報システム管理技術グループ

## 概要

IBM Cloud では、インフラストラクチャーとプラットフォームにおいて、様々なサービスが利用できる環境が提供されている。IBM Cloud ライト・アカウントの利用範囲で、Watson Assistant(IBM Watson Conversation API)を利用して、会話をする感覚で対話を通じた情報提供を行う仕組みと、WordPress API、Slack のメッセージング・プラットフォームとの連携について紹介する。

## 1 IBM Cloud とは

### [特徴]

- ・ リレーショナル・データベースから NoSQL まで、幅広いデータベース・サービスを提供。
- ・ データの完全性、低レイテンシー、並列処理が必要とされるような AI 集約型のワークロードを強化。
- ・ Watson と機械学習の幅広い API を提供することで、ユーザー独自のアプリケーション構築を支援。
- ・ 高度なセキュリティにより、ネットワーク上のデータの移動を保証。
- ・ パブリック、プライベート、および管理のトラフィックに対して、別々のネットワーク・インターフェイスを経由し、トラフィックを分離して保護することで、管理を簡素化。
- ・ ベア・メタル、コンテナ、サーバーレス・コンピューティングからハイパー・スケール・インフラストラクチャー、ブロックチェーン、AI/機械学習、HP まで、170 以上のサービスを提供。
- ・ オープン・ソースに基づいた基盤により、アプリケーションを Kubernetes、Cloud Foundry、OpenWhisk、Spark、Jupyter、R、CouchDB などで開発する選択肢を提供。
- ・ VMware、SAP、Oracle、または従来のワークロードを、容易に、より迅速にクラウドへ移行。
- ・ ローカル・ホスティングから完全なグローバル化、さらには複数のクラウド間での仲介まで、クラウド・プロバイダーを変更することなく利用範囲を拡大。

## 2 Watson Assistant について

### [特徴]

- ・ 自然言語で対話可能なアプリケーションを、シンプルな開発ツールで迅速に構築。
- ・ チャットボット等、ユーザーとコンピューターが自然言語で対話可能なアプリケーションを簡単に開発するためのサービス。
- ・ 機械学習の適用により、ユーザーからの自然言語での入力を理解し、適切な応答を返す。
- ・ 開発者はシンプルで洗練された統合開発ツールを用いることで、対話の流れを直感的な操作で作ることができる。
- ・ さまざまな業務に特化した事前定義済みのコンテンツを活用することで、より迅速なアプリケーション開発を実現。

#### [用途]

- ・ オペレータが対応していた顧客や社内ユーザーからの問合せと手続きを、チャットボットが代行。
- ・ Slack、Facebook Messenger、LINE 等のさまざまな自然言語インターフェースのバックエンド・システムとして利用。
- ・ 音声インターフェースと組み合わせ、音声自動応答 (IVR) やロボット等の機械と人間の音声対話を実現。

#### [ユーザーからの入力]

- ・ 自然言語での会話テキスト。

#### [サービスからの出力]

- ・ 入力された会話テキストから抽出されたインテント、エンティティ、及び会話フローに沿った応答文。

### 3 IBM Cloud のフリー・アカウント作成と Watson Assistant による Workspaces の作成

#### 3.1 IBM Cloud のフリー・アカウント作成

- (1) IBM Cloud [<https://console.bluemix.net/>]から、[フリー・アカウント]を作成する。
- (2) カテゴリー中、ライトプランのインフラストラクチャー、プラットフォームが利用可能。

#### 3.2 Watson Assistant による Workspaces の作成

- (1) プラットフォームの Watson から、Watson Assistant を選択することで利用ができる。
- (2) Workspaces から、Intent、Entity、Dialog と、必要に応じて、Context を設定後、会話フローを作成し、作成した会話フローを標準提供の API を使用して画面に表示する。この API の I/O は、インプットが「ユーザーの発言」、アウトプットが「ユーザーの発言に対する返事」となり、チャット画面でユーザーが発言すると、その文字列が API で POST され、返された「返事」をチャット画面に表示するという構成を取ることで、チャットボットが実現できる。
- (3) 情報提供の内容は、対応可能な情報系業務内容を例に、作成をした。
- (4) Intent、Entity、Dialog、Context について。

##### [Intent (発言の意図)]

ユーザーが発言した入力テキストに含まれる「意図」。Watson Assistant は入力テキストを分類器で分類し、その発言でユーザーが求めている意図を理解する。

(例) #あいさつ おはよう、おはようございます、こんにちは、こんばんは

##### [Entity (意味付けされた単語)]

ユーザーが発言した入力テキストから抽出される、意味付けされたキーワード。入力テキストから抽出された Entity は、ユーザーの「目的」の対象物を示す。

(例) @車 マイカー、自家用車、自動車

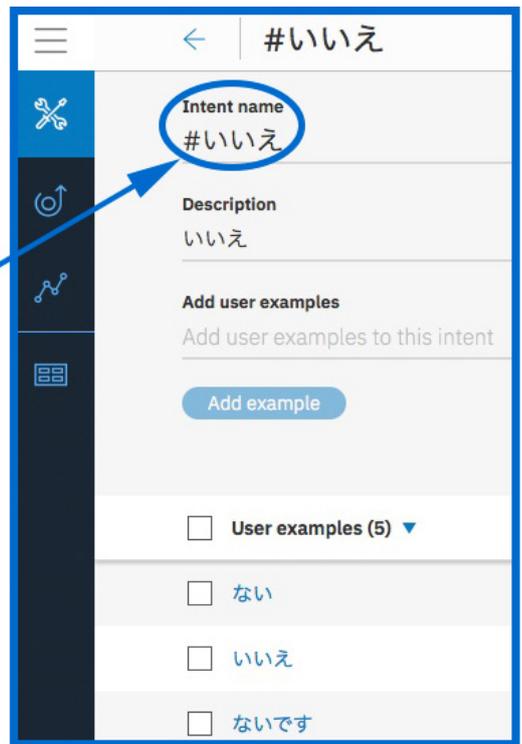
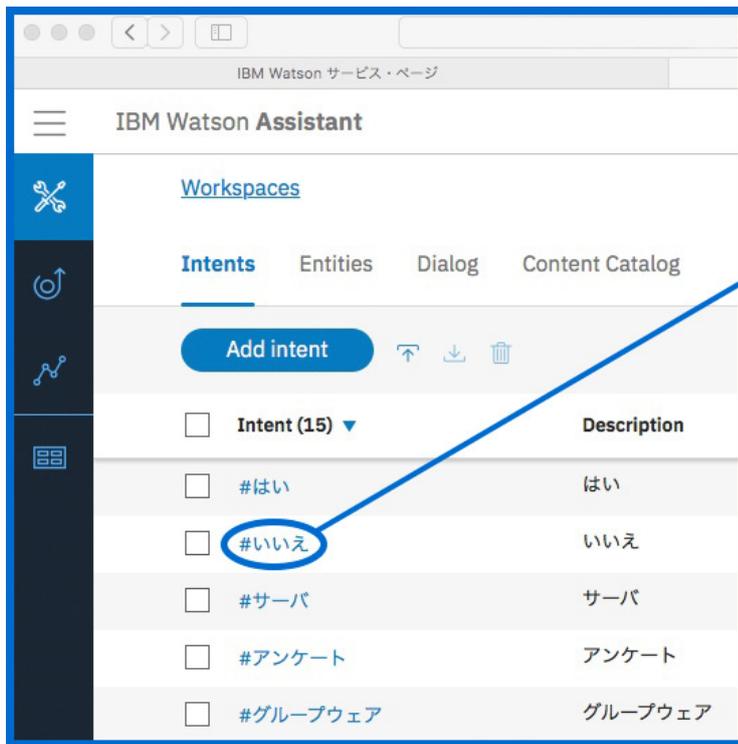


図 1. Intent

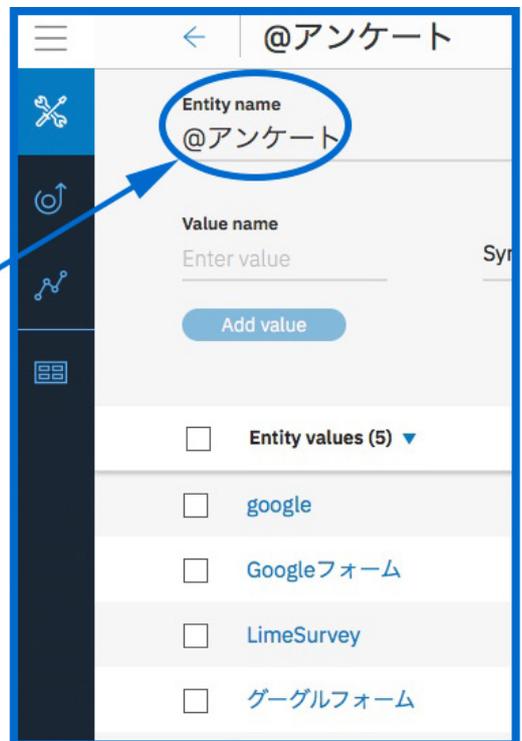
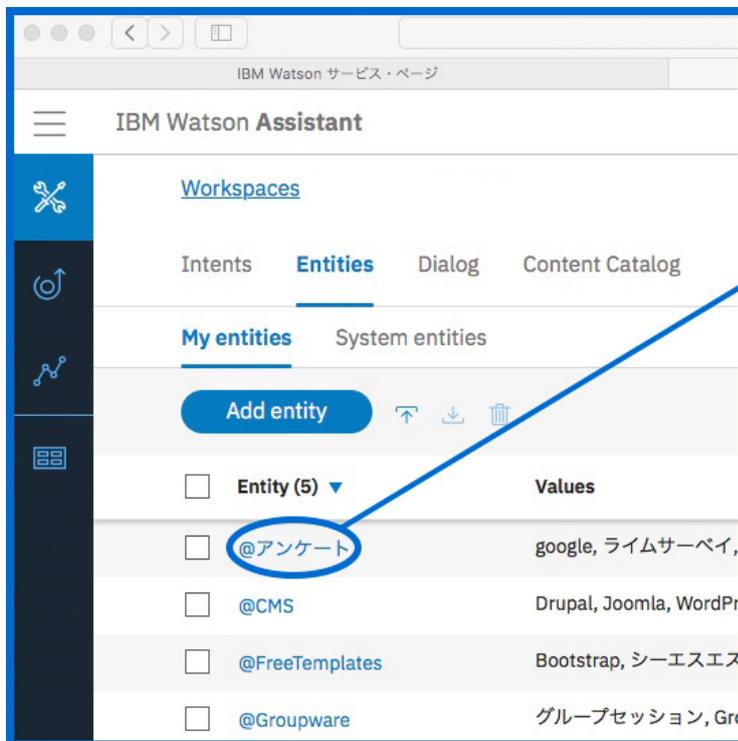


図 2. Entity

### [Dialog (対話の流れ)]

ユーザーの発言に対応して対話の流れを制御するロジックを、Dialog の組み合わせ(ダイアログ・ノード)で実装する。ユーザー要求(Intent と Entity)に応じて、ひとつの返事を返すだけの簡単なノードから、ユーザーが求めるものを掘り下げるための、木構造となったダイアログ・ノードもある。



図 3. Dialog

### [Context (会話の背景)]

ユーザーと Watson Assistant が対話を行うにあたって、情報を保持しておくことができる領域のことで、対話の回数の保持や、必要な情報がすべて入力されたかのチェックなどに利用できる。また、チャットボットが画像を返したり、DB の参照を行ったりなどのサーバー側の処理が必要な場合は、この Context 領域に任意の値をセットしておき、その値を変化させることで対応できる。

## 4 Watson Assistant から WordPress と Slack への連携

### 4.1 WordPress への連携

- (1) WordPress へ Watson Conversation WordPress プラグインをインストール後、Watson Assistant の資格情報を設定することで、WordPress のサイトにチャットボット機能が追加される。



図 4. WordPress

### 4.2 Slack への連携

- (1) Watson Assistant からのデプロイにより、メンバー間コミュニケーションツールの Slack への連携も利用できる。



図 5. Slack

## 5 まとめ

機械学習など、他の Watson サービス、AI スピーカーとの連携による音声会話を利用することで、より効果的な情報提供も可能になる。Workspaces の作成において、どのような会話情報に利用できるのかは、思索が必要になるが、一般に web サイトなどで公開されている情報は、検索により、ある程度の情報収集が可能なので、web 情報として掲載されていない情報で、問い合わせの可能性が高い有益情報を提供することで、利便性が確保されると思われる。

## 参考文献

- [1] IBM Cloud (<https://www.ibm.com/cloud-computing/jp/ja/>)
- [2] Watson とは? (<https://www.ibm.com/watson/jp-ja/what-is-watson.html>)
- [3] Watson Assistant (旧 Conversation) (<https://www.ibm.com/watson/jp-ja/developercloud/conversation.html>)
- [4] チャットボット開発を手早く簡単に (<https://www.ibm.com/watson/jp-ja/how-to-build-a-chatbot/>)
- [5] IBM Watson でコーディングのできないド素人がチャットボットを作ってみた  
(<https://www.ibm.com/think/jp-ja/watson/chatbot-conversation-1/>)
- [6] コーディングのできないド素人がチャットボットを作ってみた：ワードプレスのブログに実装編  
(<https://www.ibm.com/think/jp-ja/watson/chatbot-conversation-4/>)