**Synthèse: Performances wit.ai/Flybot**

L’outil existant est paramétré avec **7 entités:** 4 entités “personnalisées” (définies par le client), et 3 entités prédéfinies. Les entités utilisées sont les suivantes:

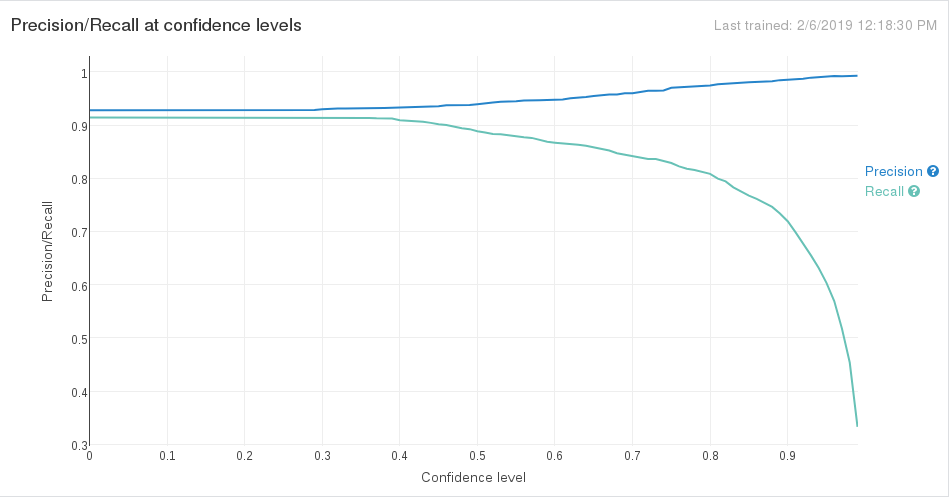
* **search\_type**: user defined, valeurs possibles: (one\_way, round\_trip), recherche par mots-clés et par texte libre
* **intent**: user defined, intention du message ou d’une partie du message, valeurs possibles: (hi, thanks, in\_progress, my\_name, etc), recherche par mots-clés et par texte libre
* **discover\_category**: user defined, type/nature/souhait du voyage, valeurs possibles: (south-america, asia, sun, romantic, visit, etc), recherche par mots-clés uniquement
* **action:** user defined, action que souhaite entreprendre l’user, valeurs possibles: (stop, restart, stop\_subscription, contest, etc), recherche par mots-clés uniquement
* **wit/datetime:** date, deux rôles possibles: départ et retour
* **wit/location:** texte libre représentant un lieu, un endroit connu ou une adresse. Deux rôles: origine et destination
* **wit/number:** extraction de chiffre à partir de free text, deux rôles: flexibility et passenger

wit.ai propose trois stratégies de recherche:

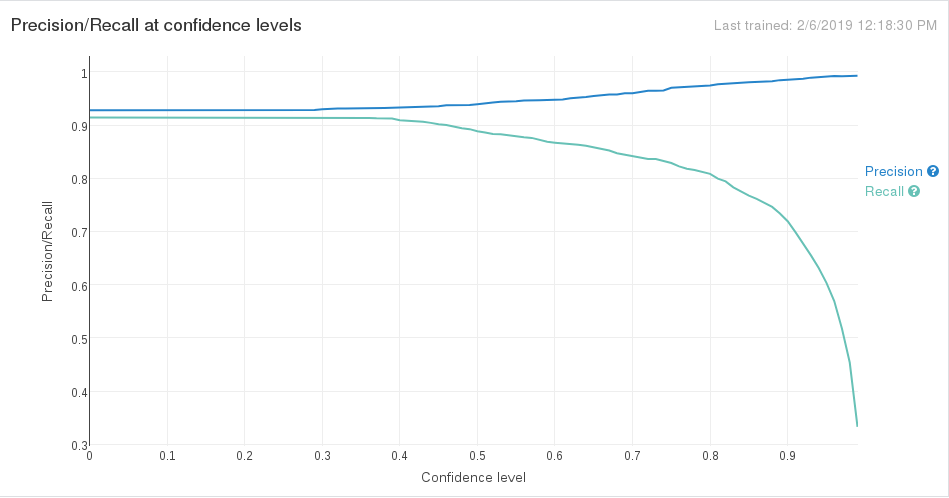
1. **Trait (previously known as Spanless)**: When the entity value is not inferred from a keyword or specific phrase in the sentence. There is no obvious association between certain words in the sentence and the value of the entity, but rather you need the sentence as a whole to determine the value.
2. **Free Text**: When you need to extract a substring of the message, and this substring does not belong to a predefined list of possible values.
3. **Keywords**: When the entity value belongs to a predefined list, and you just need substring matching to look it up in the sentence.

==================================================================

Pour les entités définies par l’utilisateur (de wit) avec une stratégie de recherche ‘free-text’, wit fourni un bilan des performances selon deux métriques: **Précision et Recall**, en fonction du  **confidence rate** de chaque entités détectée. Ci-dessous le bilan pour les deux entités “search\_type” et “intent”:



Le graphe ci-contre montre l’évolution des deux métriques en fonction du taux de confiance pour l’entité “intent”.



Le graphe ci-contre concerne l’entité “search\_type”.

Pour les deux entités, la précision est très bonne: lorsque le chatbot détecte une entité, il se trompe très peu.

Cependant, le recall est très faible lorsque le taux de confiance est élevé: lorsque des entités sont présentes dans un texte, le chatbot les détecte rarement.

**Une explication possible de ce phénomène:**

La précision augmente signifie que le chatbot se trompe très peu dans la classification des entités. Or, cela signifie aussi que, avec une précision élevée proche de 1, le chatbot ne va pas classer une entité “s’il n’est pas certain de la bonne classification”. Cela induit donc un **recall** élevé.

==================================================================

**Bilan**

* Les entités choisies sont significatives mais peuvent être raffinées: plus de précision dans les noms, plus d’entités pour couvrir le maximum d’éléments reçus de la part d’utilisateurs du chatbot
* Le recall est très faible: il faut plus d’entrainement (plus de données d'entraînement) . En particulier, plus d'entraînement pour chaque entités (plus de mots-clés pour les keywords et de logs structurés)
* Il est possible d’atteindre les mêmes performances de wit.ai avec RASA en effectuant une migration wit - RASA. L’amélioration se fera à partir de là. Pour ce faire, un premier jeu de données structuré (pas nécessairement en lien avec le domaine du tourisme) sera préparé pour tester les deux chatbots et mesurer les performances (accuracy + recall). Ces mesures permettront de décider du meilleur chatbot.