

Picovoice Rhino

PICOVOICE	Speech-to-Intent engine	
	Exigences système :	Mutiplateforme
	Développé par :	Picovoice
Licence commerciale	Personne de contact :	Bert.Vanhalst@smals.be

Fonctionnalités

[Picovoice](#) est une entreprise canadienne qui propose une plateforme permettant d'ajouter des fonctions vocales à des logiciels. La plateforme comprend un certain nombre de composants destinés, entre autres, à la transcription, la reconnaissance vocale en temps réel et la reconnaissance de l'intention basée sur la parole.

Le composant que nous abordons dans cet article s'appelle [Rhino](#), un *speech-to-intent engine* (moteur de reconnaissance vocale) permettant aux utilisateurs d'interagir oralement avec une application. La reconnaissance de l'intention d'un utilisateur basée sur une question orale s'opère normalement en deux temps : la question orale est d'abord convertie en texte, puis la reconnaissance de l'intention est exécutée sur ce texte. Rhino, pour sa part, exécute la reconnaissance de l'intention en une seule étape, directement de l'audio à l'intention, revendiquant ainsi une plus grande précision et de meilleures performances que la combinaison classique de la reconnaissance vocale (ASR, Automatic Speech Recognition) et de la reconnaissance du langage naturel (NLU, Natural Language Understanding).

Une caractéristique intéressante de la solution est sa prise en charge de différents modèles de déploiement. Elle peut être exécutée non seulement dans un cloud et on-premise, mais aussi localement sur un appareil, ce qui favorise à la fois la confidentialité des données et les performances.

De nombreux kits de développement logiciel sont disponibles pour intégrer la solution dans votre propre application, y compris pour Nodejs et Python.

Conclusions & Recommandations

Dans la pratique, en raison des grandes limitations sur le plan de la reconnaissance de l'intention et de la reconnaissance de paramètres (slots), Picovoice Rhino ne s'avère malheureusement utile que pour des scénarios très simples avec des commandes vocales fixes spécifiques. Néanmoins, l'outil présente une approche intéressante de la reconnaissance de l'intention basée sur l'entrée vocale. Le *speech-to-intent engine* fonctionne très bien et est disponible dans plusieurs langues, dont le néerlandais, le français, l'anglais et l'allemand. L'outil peut être déployé n'importe où et fonctionne également hors ligne, un atout dans le contexte de la protection des données et de la vie privée.

Tests & Résultats

À des fins de test, nous utilisons la version gratuite de Rhino. Comme scénario, nous choisissons une déclaration Dimona qui représente une intention et peut contenir deux slots (paramètres) : une date (date) et un nom de personne (worker). Il apparaît immédiatement que le nombre de [types de slots](#) que l'outil peut reconnaître par défaut est très limité. Une date (date) et une heure (heure), par exemple, ne sont pas prises en charge, alors qu'il s'agit de quelque chose d'assez basique. Par défaut, seuls les slots pour les pourcentages, les lettres, les combinaisons alphanumériques ainsi que les nombres/ordinaux d'un ou deux chiffres sont pris en charge. Nous sommes donc obligés de définir des slots nous-mêmes pour notre cas : une *date* avec comme valeurs possibles "vandaag", "morgen" et les jours de la semaine ; et un *worker* avec comme valeurs possibles "Jan", "Piet" et "Paul". Au centre de la capture d'écran ci-dessous se trouvent les exemples de phrases dans lesquels des slots peuvent être insérés. Nous pouvons tester le modèle immédiatement. En bas à droite figurent l'intention et les slots reconnus lorsque nous prononçons la phrase "een aangifte voor Jan voor vandaag".



The screenshot shows the Smals interface. On the left, there is a sidebar with 'Intents' containing 'dimona', 'Slots' containing 'date' and 'worker', and 'Macros'. The main area displays a list of phrases with highlighted slots: 'ik wil een aangifte indienen', 'een aangifte voor \$worker:worker voor \$date:date', 'een aangifte voor \$worker:worker', 'een aangifte voor \$date:date', 'een aangifte', and 'een nieuwe aangifte'. On the right, there is a 'Mic' interface with a microphone icon and a 'Click to test' button. Below the microphone, the 'Inference' results are shown: { intent: "dimona", slots: { worker: "Jan", date: "vandaag" } }.

Nous avons choisi le néerlandais comme langue du projet. Au total, [16 langues](#) sont prises en charge, dont le néerlandais, le français, l'anglais et l'allemand.

Lors des tests, nous constatons que la reconnaissance de l'intention n'est pas sophistiquée : il faut spécifier explicitement les différentes variantes des exemples de phrases pour une bonne reconnaissance. Il faut donc vraiment rendre facultatives toutes les parties optionnelles d'un exemple de phrase. Ainsi, sur la base de l'exemple de phrase "Je veux introduire une déclaration", "introduire une déclaration" n'est pas reconnu. L'outil propose bel et bien une syntaxe pour indiquer les parties optionnelles et les choix dans une liste, mais la reconnaissance de l'intention n'est donc pas flexible. L'outil semble donc plutôt utile pour des commandes spécifiques. Pour une reconnaissance flexible de l'intention, il faut recourir à une plateforme NLU plus complète. La performance est néanmoins positive, la reconnaissance est très rapide.

Conditions d'utilisation & Budget

Picovoice peut être utilisé gratuitement, avec une limitation pour Rhino à 3 utilisateurs actifs par mois et un support limité à la communauté GitHub. Le plan Developer offre jusqu'à 100 utilisateurs actifs par mois et 6 heures de support technique, et coûte 500 \$ par mois. Enfin, il existe également un plan Entreprise pour les "large scale deployments". Retrouvez plus d'informations sur [la page des tarifs](#).