Suivi de la qualité de modèles NLP

paris.py - 25 juin 2019

guillaume@clustaar.com
CTO

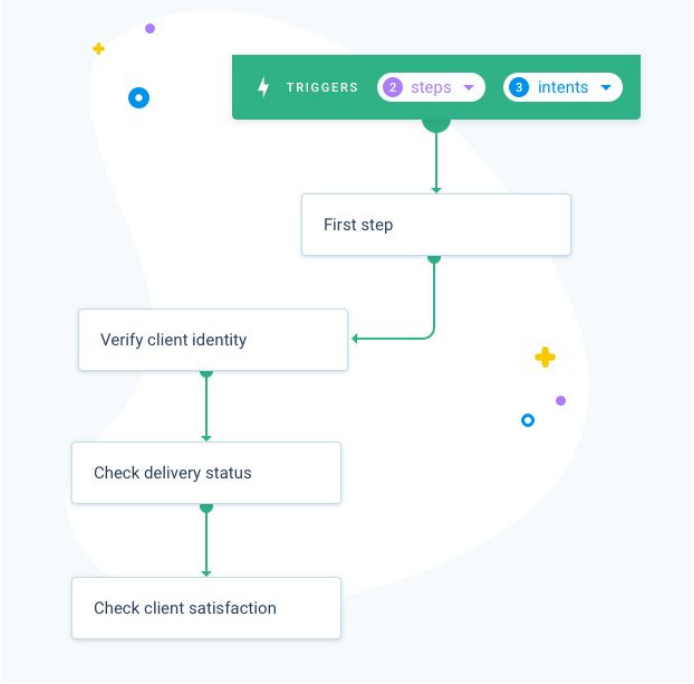
quentin@clustaar.com
Data Scientist

Contexte

Clustaar, plateforme d’automatisation du support

- Construction de bots
  - scenarii complexes, via une interface ergonomique
  - intégrations poussées (API, Webhook)

- Organisation du support
  - création de tickets
  - reprise en main synchrone ou asynchrone



Contexte

Clustaar, plateforme d’automatisation du support

Les questions “texte” sont traitées par notre moteur de **N**atural **L**anguage **P**rocessing.

Adapté pour les environnements avec :

- peu de données d’apprentissage (intents avec peu de formulation)
- des questions qui peuvent être très courtes (chat)
- des questions qui peuvent être très chargées en fautes de frappe ou d’orthographe

Problématique

Nombreuses étapes dans la construction du modèle, et la détection d'intentions.

Chaque changement susceptible de détériorer la qualité des modèles :

- Construction du vocabulaire de référence
- Spellcheck
- Normalisation
- Détection des entités
- Vectorisation
- Calcul des distances & scores
- … etc….

Problématique

QA

Il est nécessaire de mettre en place une plateforme de **Q**uality **A**ssesment pour valider les changements, et s'assurer qu'ils **améliorent** le moteur de NLP.

Construire un corpus de référence

Construire un corpus de taille significative en évitant de biaiser les résultats:

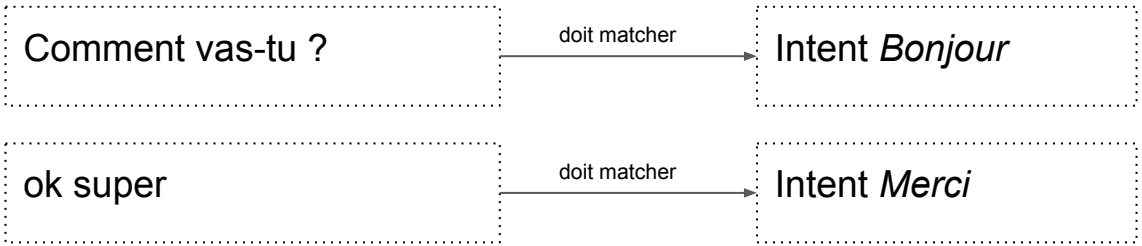
- avoir des personnes différentes qui participent au corpus (limiter la subjectivité)
- avoir des règles à respecter (*e.g* nombre d'exemples minimum)

et **figer ce corpus pour toujours**.

Construire un corpus de référence

Dans notre cas, cela signifiera :

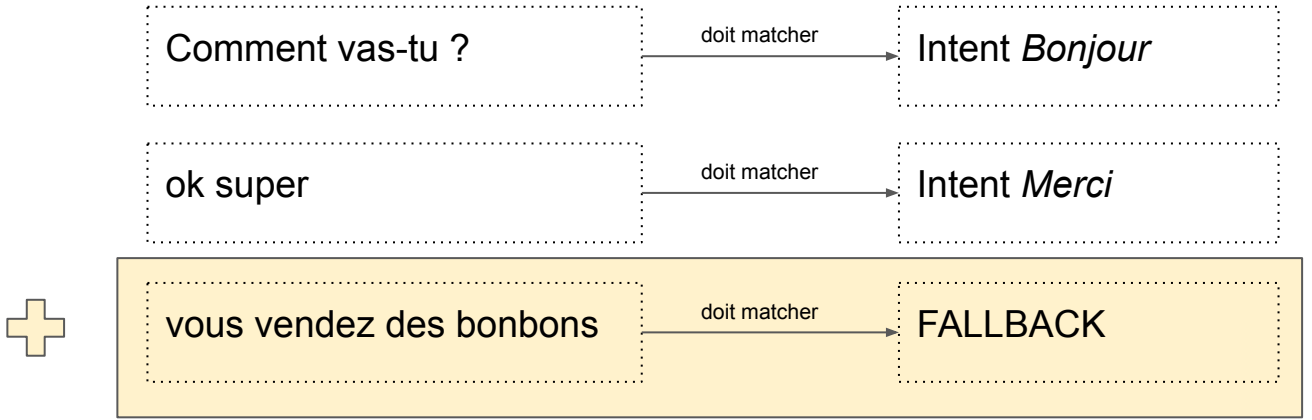
- Construire un ensemble de “faux” bots pour chaque langue
- Construire pour chacun de ces faux bots :
  - une liste de questions à poser
  - pour chaque question, le bon intent qui doit être matché
  - plusieurs questions par intent



Construire un corpus de référence

Dans le cadre d'un bot, on veut aussi tester sa capacité à admettre **qu'il ne sait pas répondre à la question.**

→ ajout d'un "faux intent" FALLBACK



Calculer des métriques de qualité fiables

Pour chaque query testée, l'ensemble des intentions du bot sont candidates, ainsi que le fallback.

$$Query_{a\ candidates} = [intent_1, intent_2, \ldots, intent_x, \ldots, intent_n, Fallback]$$

La réponse du Bot détermine alors **une prédiction pour chacun des intents**, ainsi que le fallback :

- Bien détecté : l'intention a été détectée, à raison
- Bien refusé : l'intention a été refusée, à raison
- Mal détecté : l'intention a été détectée, à tort
- Mal refusé : l'intention a été refusée, à tort

Calculer des métriques de qualité fiables

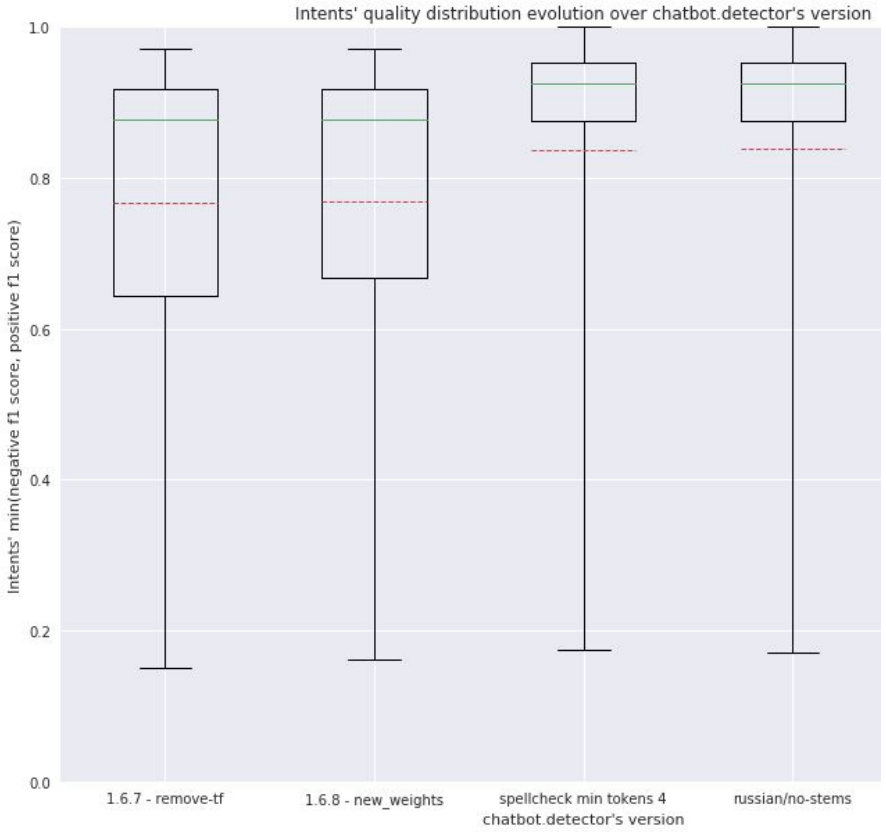
Nous avons testé différentes métriques, parmi les nombreuses possibles.

		True condition		
		Positive condition	Negative condition	
Predicted condition	Positive prediction	**True positive**	**False positive**	Positive precision = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Positive prediction}}$
	Negative prediction	**False negative**	**True negative**	Negative precision = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Negative prediction}}$
		Positive recall = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Positive condition}}$	Negative recall = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Negative condition}}$	F_1 score = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$

intention détectée correctement

intention refusée correctement

Calculer des métriques de qualité fiables



Métrique de référence :

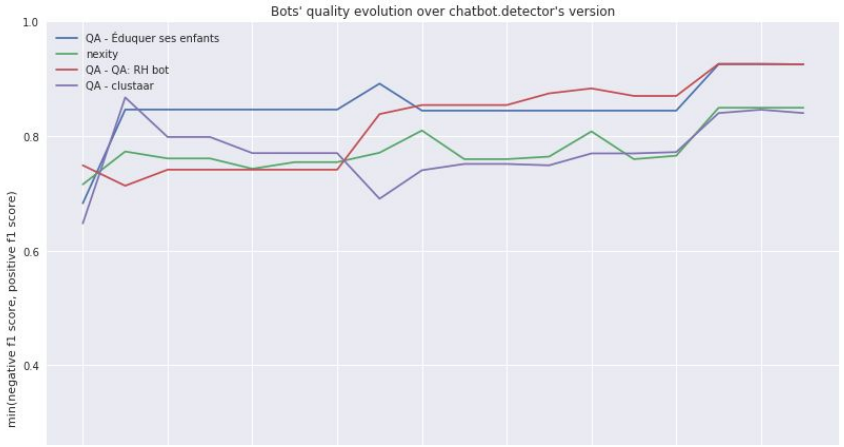
Répartition du minimum entre la $f\text{-}mesure_{positive}$ et la $f\text{-}mesure_{négative}$ pour tous les bots de QA.

(le pire des deux)

On retient là où le modèle est le plus mauvais entre :

- détecter la bonne intention
- ignorer les mauvaises intentions

Observer les résultats dans le détail



	query	detected	old_score	expected	new_score
0	quel est le rasio homme famme	Fallback	0.0	Parité ?	0.704879
1	c'est possible de travailler en double écran ?	Fallback	0.0	Matériel	0.429808
2	comment vous géré la monté en compétencd?	Fallback	0.0	Management	0.528179
3	komment vous gérez la monté en kompetence	Fallback	0.0	Management	0.528179
4	comman vous géré la montée en compétance	Fallback	0.0	Management	0.684170
5	La 'entreprise organise t'elle des conférences ?	Fallback	0.0	Conférences	0.544099

Métriques secondaires :

- évolution du min(f_{pos}, f_{neg}) par bot
- gains / pertes de “bonnes réponses” par rapport à la session précédente
- pour chaque question, la comparaison de sa réponse par rapport à la session précédente

Lancer une session de QA en Python

```
python bin/compute/compute_qa_session.py --label label-exemple --lang fr
```

```
(chatbot_qa) glebourg@gclustaar:~/dev/chatbot.qa        $ python src/chatbot/qa/bin/compute/compute_qa_session.py --l
b.com:39137/bot_platform_qa
[2019-06-17 16:10:59,549] [INFO] [qa_session_computer]: QA Session for chatbot.detector vtest-guigui fr started using
[2019-06-17 16:10:59,550] [INFO] [qa_session_computer]: Model's parameter used:
[2019-06-17 16:10:59,550] [INFO] [qa_session_computer]:         - cosine weight: 0.2
[2019-06-17 16:10:59,550] [INFO] [qa_session_computer]:         - lcs weight: 0.0
[2019-06-17 16:10:59,550] [INFO] [qa_session_computer]:         - order weight: 0.3
[2019-06-17 16:10:59,551] [INFO] [qa_session_computer]:         - subset weight: 0.5
[2019-06-17 16:10:59,551] [INFO] [qa_session_computer]:         - cosine threshold: 0.6
[2019-06-17 16:10:59,552] [INFO] [bot_trainer]: Train bot [5a579135acf861483a04d575]
[2019-06-17 16:11:23,518] [INFO] [bot_trainer]: Train completed
[2019-06-17 16:11:23,518] [INFO] [bot_session_computer]: BotSession for bot [5a579135acf861483a04d575] started
[2019-06-17 16:11:23,815] [INFO] [bot_session_computer]: Perform detections for bot [5a579135acf861483a04d575]
[2019-06-17 16:12:20,742] [INFO] [bot_session_computer]: 335 queries evaluated
[2019-06-17 16:12:20,744] [INFO] [bot_session_computer]: Prepare 21 intents results
[2019-06-17 16:12:26,519] [INFO] [bot_session_computer]: BotSession for bot [5a579135acf861483a04d575] completed
```

```
[NbConvertApp] Converting notebook /home/glebourg/dev/chatbot.qa/src/chatbot/qa/bin/viz/Languages-Sum-Up.ipynb to html
[NbConvertApp] Executing notebook with kernel: python3
```

Rapports générés dans des notebooks

Explorables facilement !

Prendre des décisions & Rassurer

accepter, **améliorer** ou **abandonner** ? C’est la Data qui parle !

Vous pouvez **prouver** l’évolution positive de la qualité de votre moteur NLP, et **rassurer** les stakeholders et les utilisateurs finaux.

Mise en place d'un outil de suivi de la qualité de modèles NLP

paris.py - 25 juin 2019

guillaume@clustaar.com / quentin@clustaar.com