L'INTÉRÊT DE COUVERTURE DE TESTS AVANCÉE POUR BCI

Cas pratique
BCIpy

ift 3913 angelo adragna 2024/2025

PROGRAMME

- Prérequis : couverture de tests avancée
- Concept de BCI
- Projet BCIpy de CAMBI
- Code, application pratique et généralisation
- Synthèse
- Devinette

PRÉREQUIS COUVERTURE DE TESTS AVANCÉS

Couverture de tests

&

Couverture de tests avancés

- la couverture de code : le rapport entre nombre de lignes de codes exécutées par les tests et le nombre de lignes de codes total = Métrique et déterministe.
- L'analyse de mutation : Introduction variation du code censée provoquer une erreur pour montrer qu'un test est capable de détecter des erreurs = Mise en lumière de tests inefficaces.
- La couverture de scénarios : On vérifie que la plupart des scénarios sont testés. Scénario est un chemin fonctionnel ou cas d'utilisation.
- La couverture d'exigences : On vérifie que la plupart des exigences sont testées. Exigence est une description d'une fonctionnalité. Comme dans un cahier des charges.

VULGARISATION DU CONCEPT DE BCI (BRAIN COMPUTER INTERFACE)

SIGNAUX CÉRÉBRAUX

Capture

- Capture signaux (EEG)
- Filtrage et prétraitement (suppression bruit)
- Analyse des signaux
- Interprétation des intentions
- (apprentissage automatique)
- Envoi des commandes du dispositif

Commandes

ORDINATEUR:
machine
robot
prothèse

Article : https://www.ncbi.nlm.nih.gov

Ref : https://www.thedifferentialdx.com C3PO gif

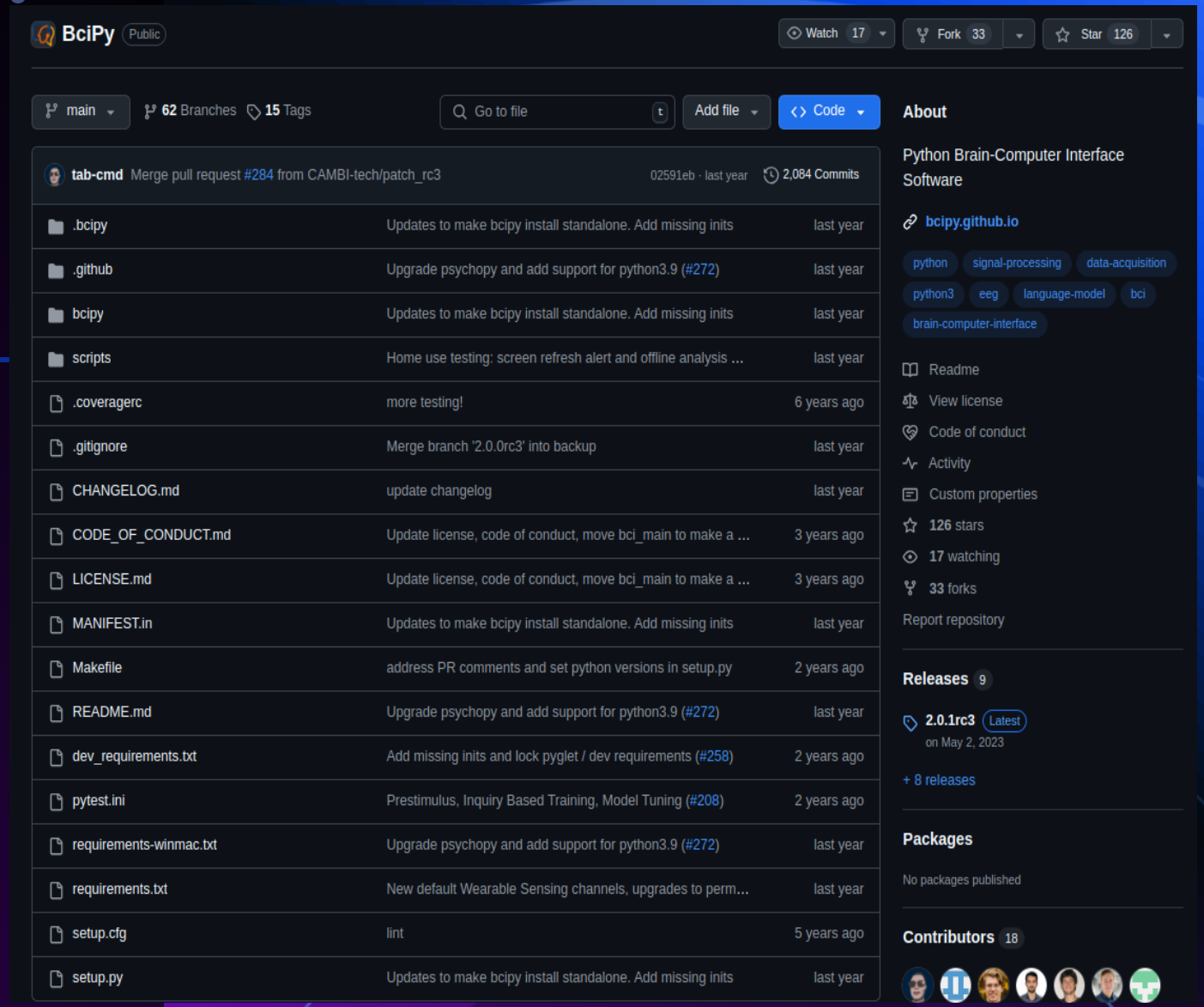
BCIpy par CAMBI

Groupe de chercheurs CAMBI.

Librairie fonctionnant comme application autonome et open source.

Réaliser des expériences d'interface cerveau-machine en python.

https://github.com/CAMBI-tech/BciPy



Code, pratique et théorie : Couverture de scénarios

- Test "prédict" de gpt2 sans contexte : liste vide en entrée, on récupère les probabilités puis on les analyse.

```
112     def test_gpt2_predict_start_of_word(self):
113         """Test the gpt2 predict method with no prior evidence."""
114         symbol_probs = self.gpt2_model.predict(evidence=[])
115         probs = [prob for sym, prob in symbol_probs]
116 
117         self.assertTrue(
118             len(set(probs)) > 1,
119             "All values should not be the same probability")
120         # Consider whether the following assertion should be True
121         # backspace_prob = next(prob for sym, prob in probs if sym == BACKSPACE_CHAR)
122         # self.assertEqual(0, backspace_prob)
123         for prob in probs:
124             self.assertTrue(0 <= prob < 1)
125         self.assertAlmostEqual(sum(probs), 1, places=3)
```

- Pratique :

Ce code s'assure d'avoir les prédictions valides sans contexte au préalable. En pratique si les prédictions sans contexte étaient "parfaites" la main pourrait se resserrer par reflexe quand la personne sursaute.

- Théorie :

Sans couverture de scénarios, on entrave l'anticipation des comportements réels, tel que l'etat mental et physiologique.
Donc provoquer des mouvements incorrects ou non désirés.

Code, pratique et théorie : Couverture d'exigences

- Ce code test le constructeur CausalLanguageModel de gpt2 : pour s'assurer de lever une exception UnsupportedResponseType lorsque le type de réponse fourni n'est pas supporté par le modèle.

```
51      def test_unsupported_response_type(self):
52          """Unsupported responses should raise an exception"""
53          with self.assertRaises(UnsupportedResponseType):
54              CausalLanguageModel(response_type=ResponseType.WORD,
55                                  symbol_set=alphabet(), lang_model_name="gpt2")
56
```

- Pratique :

Ce test contribue à la robustesse et à la fiabilité en garantissant que la classe réagit correctement aux entrées non valides et en prévenant les erreurs potentielles à l'exécution.

- Théorie :

Sans couverture d'exigences, on s'expose à l'imprécision, au manque de robustesse et aux fonctionnalités manquantes.
Donc des mouvements manquants ou non pris en charge.

~~Code, pratique et~~ théorie : Analyse de mutation

- Théorie :

Sans analyse de mutation, on pourrait laisser passer des prédictions moins précises en grande partie à cause de tests inefficaces. Donc des mauvaises interprétations peuvent provoquer des mouvements saccadés (lag et latence).

SYNTHESE

- Exemples physiques et concrets : pour montrer intérêts de la couverture de code avancée.
- Plus qu'une métrique.
- Causer des réels problèmes, dans le BCI comme ailleurs.

Le BCI c'est quand même la classe !

DEVINETTE

https://www.cambi.tech/about

COMMENT RECONNAIT-ON DES SCIENTIFIQUES SEULEMENT AVEC UNE PHOTO ?