Dominic JOBIN   
Dominic MICHAUD  
Groupe 0001

**Outils d’expérimentation**  
  
Travail présenté à M. François Bertrand  
Exploration des nouvelles technologies  
420-501-SF

Département de l’informatique  
Cégep de Sainte-Foy  
Vendredi le 17 novembre 2017

Table des matières

[Démarche d’expérimentation 2](#_Toc496877167)

[Devis techniques des outils 2](#_Toc496877168)

[Exécutable, sources et documentation 2](#_Toc496877169)

# **Démarche d’expérimentation**

**Langage et environnement de programmation**

Nous utilisons le langage Python pour sa facilité d’utilisation et sa simplicité. Au départ du projet, nous avions de l’expérience dans ce langage. Nous étions convaincus qu’il y avait des moyens de créer notre projet. Pour l’IDE, nous avons choisi PyCharm parce qu’il était déjà installé sur nos postes.

**Réseau de neurones**

Nous avons commencé par chercher une librairie qui permet de créer un réseau de neurones. Nous avons essayé PyBrain, Theano et NeuroLab. Nous avons abandonné rapidement Theano parce qu’il est trop compliqué à utiliser. Nous avons utilisé PyBrain au départ.   
  
Nous lui avons donné des données et des directives au réseau de neurones. La première directive était de lui faire apprendre 36 caractères dans un cerveau. Cela a échoué car il avait trop d’élément à apprendre en même temps. La deuxième tentative fut de créer plusieurs cerveaux qui regroupent des caractères qui se ressemblent et un qui fait le lien avec ces cerveaux. Nous avons pensé qu’avec cette façon que les caractères proposés montrent les caractères similaires. Nous croyons qu’avec cette manière de faire, que l’efficacité de l’apprentissage augmenterait à un niveau permettant d’avoir des résultats pertinents. Le réseau de neurones a fait du sur-ajustement, ce qui n’est pas ce qu’on recherche. La liste de caractères proposé montrait des caractères qui ne devrait pas être suggéré (exemple : l’utilisateur dessine un « 8 » et le second choix proposé est un « M »). Nous avons tenté de faire plusieurs cerveaux qui regroupent des caractères qui se diffèrent. Dans cette expérimentation, il y a moins de sur-ajustement, mais la reconnaissance des caractères est faible (exemple : l’utilisateur dessine un « R » et le premier choix proposé est un « K » et le « R » est suggéré en quatrième choix). Nous avons changé pour NeuroLab et ressayer ces deux manières d’apprentissage. Avec les cerveaux qui regroupent des caractères similaires, nous avons eu un résultat qui correspond au ce qu’on a souhaité. L’expérimentation était différente à ce qu’on a imaginé surtout à cause qu’on ne connait pas la manière que PyBrain effectue son apprentissage.

**L’interface graphique et la grille**

Nous avons utilisé Pygame et Image pour l’interface graphique. Ce fut plus facile pour nous de faire la grille et les interactions. La première dimension qu’on a donnée à la grille était de 48 carrés par 48 carrés. Nous avons rencontré des problèmes de performances avec l’apprentissage du réseau de neurones parce qu’il y avait trop d’éléments à traiter. Nous avons diminué la dimension de la grille à 16 carrés par 16 carrés.

# Devis techniques des outils

**Librairies utilisées :**

* Pygame;

# Exécutable, sources et documentation