AS - TP 7
Réseaux récurrents : Séquence à séquence (seq2seq)

Nicolas Baskiotis - Benjamin Piwowarski

2019-2020

Introduction (brève, cf cours)

Dans ce TP, nous allons étudier l'utilisation des RNNs pour deux tâches :
— Étiquetage de chaque élément de la séquence
— Génération d'une séquence à partir d'un état latent. Cet état latent représente la donnée en entrée, et dépend de la tâche : représentation de la phrase (traduction/question réponse), représentation d'une image (légende d'image), etc.

Par rapport aux TPs précédents, deux nouveautés :

1. Utilisation de PackedSequence qui permet de traiter de manière efficace des séquences de taille variable.
2. Utilisation des RNN définis par torch qui prennent en entrée une `PackedSequence` et renvoient en sortie des structures similaires.

1 Exo 1 : Étiquetage

Dans cet exercice, nous allons nous intéresser à la tâche d'analyse syntaxique (Part-Of-Speech) qui consiste à associer à chaque mot une nature/catégorie grammaticale.

Utilisez datamaestro pour récupérer le jeu de données GSD en utilisant

```
from datamaestro import prepare_dataset
ds = prepare_dataset("org.universaldependencies.french.gsd")
train, dev, test = (ds.files[n].data() for n in ("train", "dev", "test"))
```

Chaque exemple est une phrase, déjà segmentée en tokens, pour lesquels nous nous intéresserons au mot brut (`form`) et au tag (`upostag`).

Dans un premier temps, définissez un `Dataset` où chaque item est un couple (tokens, tags) où les tokens sont les mots d'une phrase et `tags` sont les catégories associées. Puis définissez la fonction de collage `collate_fn` du `DataLoader` en utilisant.

Questions complémentaires
— (obligatoire) Tenir compte du problème des mots OOV pendant l'apprentissage
— (recommendé) Afficher pour une phrase donnée en entrée sa décomposition

2 Exo 2 : Traduction

Pour la tâche de traduction, nous allons utiliser deux RNNs :
— un encodeur qui est en charge de produire un état caché après avoir lu la séquence à traduire
— un décodeur, qui à partir de l'état caché, va engendré la phrase traduite.

En plus du token *EOS (End of Sequence)*, vous aurez besoin d'un token spécial *SOS (Start of Sequence)* qui sera le premier token donné en entrée au décodeur (en plus de l'état caché) à partir duquel la phrase est traduite. Pour l'apprentissage, deux modes sont possibles entre lesquelles il faut alterner :

— mode contraint : on passe au décodeur en entrée la phrase cible

— mode non contraint : on passe au décodeur le token précédement engendré.