PythonとKerasによるディープラーニング

第6章 テキストとシーケンスのためのディープラーニング
6.1 テキストデータの操作

19ss317u　　ZHAO　YI

6.1.3 テキストのトークン化から単語埋め込みまで

- IMDbデータをテキストとしてダウンロードする
  1)IMDbデータセットを元の状態でダウンロードし、展開する。
  2)訓練に使用する個々の映画レビューを文字列のリストにまとめる。
- データのトークン化
  テキストをベクトル化し、訓練データセットと検証データセットに分割する。
- GloVeの単語埋め込みをダウンロードする
  GloVeプロジェクトのWebサイトにアクセスし、2014年の英語のWikipediaのデータを使って学習した埋め込みをダウンロードする

6.1.3 テキストのトークン化から単語埋め込みまで

- 埋め込みの前処理
  1)展開したファイル(.txt)を解析し、単語(文字列)をベクトル表現(数値ベクトル)にマッピングするインデックスを構築する。
  2)embedding層に読み込むことができる埋め込み行列を作成する。
  注意点：インデックス0はプレースホルダであり、単語やトークンを表さないことである。
- GloVeの埋め込みをモデルに読み込む
  1)GloVe行列の準備ができたら、このモデルの層であるembedding層に読み込む。
  2)embeddingのtrainable属性をFalseに設定することで、embedding層を凍結する。

6.1.3 テキストのトークン化から単語埋め込みまで

- モデルの訓練と評価

  1)このモデルをコンパイルし、訓練する。

  2)このモデルの性能をプロットする。

6.1.4　まとめ

- テキストをニューラルネットワークで処理できるものに変換する。
- Kerasモデルのembedding層を使って、タスクに特化したトークン埋め込みを学習する。
- 学習済みの単語埋め込みを使って、簡単な自然言語処理問題での性能をさらに向上させる。