UNIVERSIDADE FEDERAL DO PIAUÍ - UFPI
Centro de Ciências da Natureza - CCN
Departamento de Computação - DC
Programa de Pós-Graduação em Ciência da Computação
Disciplina: Tópicos em Sistemas Computacionais
Professor: Raimundo Santos Moura
Alunos: José Patrício de Sousa Filho

RELATÓRIO DE ATIVIDADE PRÁTICA

Os classificadores foram desenvolvidos e treinados utilizando o classificador Naive Bayes componente do NLTK. Cada diretório contém um arquivo chamado `generate_classifier.py` que quando executado por meio do comando `python enerate_classifier.py` de dentro do diretório do classificador é responsável por carregar o *corpus*, treinar o classificador, exibir o valor da acurácia obtida e salvar o classificador no arquivo `classifier.pkl`.

No diretório de cada classificador está presente também um arquivo chamado `classifier.py`, este pode ser utilizado através da execução do comando `python classifier.py input.txt`, sendo o *input.txt* um nome qualquer para o arquivo contendo o texto de entrada a ser classificado. O `classifier.py` utiliza o `classifier.pkl` gerado pelo `generate_classifier.py` para que não seja necessário fazer novamente o treinamento do classificador, ganhando assim velocidade na utilização.

Em alguns dos classificadores foi necessária a utilização de outras bibliotecas(para *parsing* de HTML e outras tarefas), então faz-se necessária a instalação desses pacotes adicionais através do comando `pip install -r requirements.pip`.

1. Classificador de Documentos

O classificador de documentos foi feito utilizando um corpus que contém aproximadamente 2000 letras de músicas das mais buscadas do site www.letras.mus.br. Os documentos estão classificados de acordo com o estilo musical de cada composição, sendo eles: gospel e funk. O *script* utilizado para *download* das letras está salvo no arquivo `download_lyrics.py`.

Na construção do classificador diversas *features* foram testadas afim de melhorar a acurácia do modelo. Dentre as testadas foram escolhidas as seguintes:

1. Proporção entre a quantidade de palavras repetidas e a quantidade total de palavras no texto;

2. Tamanho médio das palavras;
3. Número de palavras;
4. Quantidade de palavras positivas e negativas obtidas através da utilização do léxico de sentimentos Sentilex-PT;
5. Número de palavras presentes no texto e não encontradas no Sentilex-PT;
6. Proporção entre a quantidade de vezes que cada classe gramatical é encontrada no texto e a quantidade total de palavras do texto;

O classificador conseguiu acurácia de 92%.

2. Segmentador de Sentenças

O segmentador de sentenças foi treinado utilizando o *corpus* Floresta componente do NLTK e possui acurácia de 95%. O programa imprime o texto de entrada com as *tags* <SEGMENTADOR> e </SEGMENTADOR> ao redor dos segmentadores de sentença encontrados na página.

3. Identificação de Tipos de Diálogo

O classificador de tipos de diálogo foi feito utilizando um corpus que contém uma coleção de mensagens eletrônicas coletadas do sistema de atendimento de uma empresa de tecnologia. As mensagens estão classificadas nas categorias: saudação, dúvida, problema e outros.

As features escolhidas para classificação das mensagens foram:

1. Palavras contidas no texto: a verificação se a palavra W está presente no texto da mensagem;
2. Padrão de classes gramaticais do texto: para tal foi utilizado um outro classificador(já treinado e somente carregado) que com uma palavra de entrada retorna a sua classe gramatical. Este classificador foi treinado com palavras dos *corpora* Mac Morpho e Floresta componentes do NLTK.

O classificador apresentou acurácia de 65% na avaliação realizada. O desempenho pode estar relacionado à baixa qualidade da base de dados(muitas palavras grafadas incorretamente e outras formas de lixo).

4. Identificação de Rótulos

Para este classificador não foi possível encontrar um *corpus* para inferência textual em português, o que impossibilitou a criação do classificador.