Classificação e Pesquisa em Dados - INF01124 Turma A
Enunciado do Trabalho Prático
Profa. Karin Becker

1. Definição do Problema.

A análise de sentimentos visa determinar automaticamente o sentimento presente em um texto. Neste trabalho entenderemos sentimento como polaridade, a saber negativa, neutra e positiva. Seu objetivo neste trabalho é utilizar os conhecimentos adquiridos na disciplina para tratar um problema de análise de sentimento em tweets.

A solução de análise de sentimentos será baseada em um *dicionário de sentimentos*. Cada palavra tem um escore de sentimento associada, e a soma destes escores dá o sentimento do tweet. Assumiremos a seguinte escala:

- soma > 0.1: positivo
- soma < -0,1: negativo
- -0,1 >= soma <= 0,1: neutro

Por exemplo, dado os escores abaixo para três palavras contidas no dicionário, o tweet "Amo Garopa, maior astral !!" tem polaridade positiva. Note-se que nem todas palavras existem no dicionário (ex: Garopaba), quando o respectivo escore de sentimento deve ser considerado como 0.

amo = 0,7; astral:0,3; maior:-0,1
Amo Garopapa, maior astral = 0,7+0+(-0,1)+0,3=0,8 > 0.1 = positivo

Sua tarefa será:

- Criar um *dicionário de sentimentos* a partir de um conjunto de tweets rotulados;
- Determinar a polaridade de tweets novos;
- Implementar uma *segunda funcionalidade*, dentre as opções oferecidas no enunciado abaixo, as quais são uteis para entender o uso das palavras e sua relação com o sentimento.

Para isto deverá escolher duas dentre as estruturas de busca a dados vistas na disciplina, projetar e implementar sua solução. Os tweets dados de entrada devem ser armazenados em um arquivo cujo formato e forma de acesso você vai escolher e justificar. Sua aplicação deve:
a) ler tweets de uma entrada csv, criar o dicionário e demais estruturas de busca. Os tweets devem ser armazenados em um arquivo. O armazenamento do dicionário em arquivo é opcional (mas recomendado);
b) adicionar mais tweets aos existentes, adicionando novos tweets ao arquivo, e atualizando o dicionário e as demais estruturas de busca.
c) dado um conjunto de tweets sem polaridade, determinar a polaridade de cada um deles
d) realizar a funcionalidade de busca adicional escolhida.

2. Criação do Dicionário de Sentimentos

Você deve escolher uma estrutura de dados para representar seu dicionário. Lembre-se que deve considerar que:
a) seu dicionário é uma representação de um conjunto de tweets armazenados;
b) a qualquer momento, deve ser possível incluir mais tweets, resultando na atualização tanto do dicionário, quanto do arquivo com o conjunto de tweets armazenados (e de todas demais estruturas que apoiam o acesso aos dados).

Para criação/atualização do dicionário de sentimentos, o procedimento é:
Dado um arquivo .csv que será fornecido, enquanto houverem tweets não lidos:

- Ler um tweet que contém o texto e a polaridade do tweet (-1, 0 ou 1, representando negativo, neutro e positivo, respectivamente);
- extrair as palavras do tweet (desprezar pontuações, considerar apenas palavras com mais de 2 letras, converter tudo para letras minúsculas);
- Cada palavra do tweet recebe inicialmente o escore do tweet;
- Incluir cada palavra no dicionário. Se a palavra:
    - não existe: incluir a palavra e associá-la com as seguintes informações <escore do sentimento, escore acumulado, número de tweets onde foi usada>. Especificamente estes valores são <escore do tweet, escore do tweet, 1> ;
    - já existe: atualizar as informações associadas. Mais especificamente, atualizam-se os dois acumuladores (do numero de tweets e do sentimento acumulado), e calcula-se o novo escore de sentimento da palavra através da divisão do sentimento acumulado pelo nro de tweets ;
- gravar o tweet em um arquivo.

Para exemplificar, considere os tweets abaixo:

-1	**Jogo** hoje, **maior** furada
1	Saida com a turma, **amo** muito !
1	**Amo** o Rio, **maior** astral
-1	**Jogo** foi a **maior** decepção
-1	Que decepção o **jogo**, **maior** roubada.
1	**Maior Jogo** hoje!
0	Saida do Rio hoje
0	Dia de **Jogo**

Com base nestes tweets e considerando as palavras em negrito, o escore de sentimento, escore acumulado e número de tweets armazenados no dicionário ao final devem ser:
jogo → <-0,4; -2; 5>
maior → <-0,2; -1; 5>
amo → <1, 2, 2>

Dicas:
- procure a função de tokenização de sua linguagem para processar o texto dos tweets. Ela vai lhe ajudar a separar e selecionar as palavras;

- procure uma lista de "stop words" em português (o, a, os, as, de, com, etc), a fim de não criar entradas de pouco valor no dicionário. Seus resultados (do ponto de vista de análise de sentimento) vão ficar melhores. A remoção de stop words é opcional.

3. Determinação de Polaridade

Dado um conjunto de tweets em um arquivo de entrada, deve ser possível determinar a polaridade de cada um deles. O resultado deve poder ser exportado em um arquivo .csv.

4. Funcionalidade de busca adicional

Para poder melhorar o dicionário, são importantes algumas funcionalidades que permitirão entender o uso das palavras e sua relação com o sentimento. Você deve escolher uma dentre as funcionalidades abaixo, e projetar/implementar a funcionalidade utilizando uma estrutura de busca de apoio apropriada, **diferente** daquela utilizada para o dicionário.

a) Dada uma palavra, gerar um arquivo csv com todos os tweets que a contém, e a respectiva polaridade. O critério de pesquisa para esta busca são: palavra, e opcionalmente a polaridade dos tweets buscados. Usando os tweets do exemplo anterior, pode-se buscar todos os tweets que contenham a palavra *jogo* (quatro tweets), ou somente os tweets com polaridade negativa que contenham a palavra *jogo* (somente dois tweets).

b) Dado um radical de duas ou mais letras, buscar todas as suas variações encontradas nos tweets. Por exemplo, com o radical "am", poderiamos encontrar diversas conjugações do verbo amar (amei, amamos), variações do substantivo (amor, amorzinho), formas de expressar sentimento (ameeeeeeeei, amooooo), além de outras palavras não relacionadas (amazonas, americano).

c) Buscar palavras cujo escore sentimento esteja em algum intervalo (exemplos: palavras com escore de sentimento maior que 0.3, ou entre 0.1 e 0.5)

Você deve escolher uma das três funcionalidades de busca acima, e implementá-la. Deve justificar que a estrutura adotada para apoiar a busca escolhida é adequada ao problema, considerando as funções de inclusão e busca.

5. Definição dos Grupos

- O trabalho deve ser realizado em **duplas**.
- Para poder realizar em **triplas**, devem ser escolhidas **duas** funcionalidades de busca.
- O trabalho individual deve ter uma boa razão, e ser autorizado pela professora.
- A composição do grupo deve ser informada no moodle, via a tarefa definida para este fim.
- Todo o grupo deve estar presente na avaliação oral do trabalho. Caso algum colega esteja ausente, será interpretado que ele/ela não realizou o trabalho.

- Todos os membros do grupo devem ser capazes de responder sobre qualquer coisa do trabalho.

6. Material a ser entregue

- Relatório
- Código (.zip)

Tarefas serão abertas para enviar o material. Não envie seu trabalho por email.

7. Estrutura do Relatório

Seu relatório deve justificar suas escolhas, e explicar com clareza o projeto feito. Ele será avaliado pela clareza, objetividade, correção, completude, e qualidade das argumentações apresentadas, bem como da descrição do projeto/implementação. O formato do relatório é livre. Em linhas gerais ele deve conter:

a) A informação sobre os integrantes do grupo;
b) informação sobre a infraestrutura escolhida (ex: linguagem de programação, frameworks, etc)
c) a organização e método de acesso do arquivo contendo os tweets, com justificativa da escolha;
d) a estrutura de dados para representar o dicionário, a justificativa da escolha, e os procedimentos de inserção e atualização. deve ficar claro em particular como o texto do tweet foi pré-processado para inserção no dicionário;
e) descrição da funcionalidade de previsão de sentimento (como pega um tweet novo e atribui a ele uma polaridade);
f) a estrutura de dados para apoiar a funcionalidade de busca escolhida, a justificativa da escolha, e os procedimentos de inserção e atualização

8. Diretrizes

- O trabalho pode ser realizado em qualquer linguagem de programação que os alunos estimarem adequada para o trabalho.
- **Não podem** ser utilizadas
    - **bibliotecas** com funções prontas para as estruturas de dados escolhidas que podem ser apenas chamadas (e.g. classes Map, Hashmap de linguagens como Java ou Python, ou alguma classe implementando uma lista invertida, árvore B ou trie). Seu código deve incluir a estrutura de pesquisa e implementação de suas respectivas operações ;
    - **sistemas de gerência de dados** de qualquer natureza. Você deve utilizar as funções de acesso de arquivos disponíveis em sua linguagem.
- Se você pegar código de terceiros, certifique-se que você entende de sua implementação. Você será questionado por suas escolhas, e se não demonstrar familiaridade com seu código, sua implementação **não** será considerada .
- Casos de plágio serão tratados com severidade.

9. Avaliação

- Relatório: 40%
- Código: 40%
- Questionamentos : 20% (pode ser individualizado)

10. Datas

- **Informe do Grupo** (via tarefa do moodle): 18/06
- **Entrega do material** (via tarefa do moodle): 06/07 – 20% de desconto sobre a nota total do trabalho com um dia dia de atraso, 10% por dia adicional de atraso.
- **Demonstração e Avaliação Oral do Trabalho**: 11/07 e 13/07 (opcional 18/07 se necessário)