Universidade Federal de São Carlos
Aprendizado de Máquina
Prof. Tiago A. Almeida <talmeida@ufscar.br>

PROJETO – FINAL DE CURSO

Datas importantes

1. **Prazo para submissão: 11/06 – 08h00 (via AVA).**
2. **Apresentações: 17/06 – a partir das 14h00.**

Aplicação de métodos de aprendizado de máquina em problemas de processamento de linguagem natural

Processamento de Linguagem Natural (PLN) é uma subárea da Inteligência Artificial que consiste no desenvolvimento de modelos computacionais capazes de realizar tarefas baseadas em informações expressas em linguagem natural. Para isto, os modelos devem ser capazes de "entender" e processar a maneira como os humanos se utilizam para comunicar, seja através de textos ou sons.

Diferente de valores numéricos e tabelas, os dados estudados dentro da área de PLN tendem a não ser estruturados, aparecendo em diversos tamanhos e formatos diferentes.

A área possui diversos problemas conhecidos e estudados na literatura. Para este projeto, foram selecionados um conjunto de temas e problemas relacionados a área de processamento textual, presentes em aplicações e tecnologias que utilizamos em nosso cotidiano.

Os temas abordados serão:

Grupo	**Tema**
1	Detecção de Spam em mensagens SMS
2	Detecção de Spam em comentários do YouTube
3	Classificação do domínio de questões
4	Análise de sentimento em *tweets*
5	Análise de sentimento em *reviews* na Amazon
6	Detecção de *fake news* em notícias brasileiras
7	Análise de *fake reviews* de plataformas como TripAdvisor, Expedia e Yelp

Implementação

A implementação deverá ser feita exclusivamente em Python. Poderá ser feito o uso de bibliotecas de manipulação de dados utilizadas ao longo do curso, como *Pandas* e *Numpy*. Não será permitido o uso de bibliotecas de Aprendizado de Máquina ou qualquer outra implementação já existente (exceto para o SVM). O código deverá ser completamente original.

Dentre os métodos avaliados, obrigatoriamente as seguintes técnicas precisarão ser implementadas, analisadas e comparadas (**não limitado a**):

1. K-vizinhos mais próximos;
2. Regressão logística;
3. Naive Bayes;
4. Redes neurais artificiais;
5. Máquinas de vetores de suporte.

Cada grupo deverá fornecer `scripts` gerais que realizem as execuções dos métodos e análises reportadas no relatório.

Bases de dados e referências

As bases de dados e instruções estão detalhados nos *links* disponíveis nas descrições de cada problema.

Análise dos métodos e resultados

Além da implementação dos métodos, boa parte da nota do projeto será atribuída à análise realizada pelo grupo nos seguintes itens:

- Análise de diferentes formas de representação e pré-processamento de texto;
- Ajuste de parâmetros e análise de impacto;
- Curvas de aprendizado, detecção e correção de super ou sub-ajustamento;
- Análise de redução de dimensionalidade para exibição dos dados e aceleração do treinamento (se necessário);
- Análise de desempenho individual e comparação dos resultados usando modelos estatísticos.

Para que os experimentos sejam totalmente reprodutíveis, espera-se que todas as decisões tomadas pelo grupo sejam reportadas e justificadas.

Relatório

O relatório deverá ser escrito em LaTeX, usando formato IEEE (disponível no AVA), e recomenda-se que contenha as seguintes seções (veja o exemplo fornecido em `relatorioExemplo.pdf`):

1. Resumo;
2. Introdução: descrição do problema e motivação;
3. Revisão da literatura: descrição de como outras pesquisas abordaram o problema;
4. Base de dados: descrição da base de dados, atributos e pré-processamentos realizados;
5. Experimentos: descrição do sistema de avaliação (treinamento e teste), definição de parâmetros e medidas de desempenho empregadas.
6. Resultados: descrição e análise dos resultados (apresentar tabelas e gráficos).
7. Conclusões;
8. Referências: lista completa de trabalhos consultados; e
9. Apêndice: tutorial de execução dos métodos implementados e análises realizadas. Esta seção precisará fornecer o passo-a-passo necessário para obter os resultados reportados.

O relatório poderá conter no máximo 6 páginas em coluna dupla (sem contabilizar o Apêndice).

Interface

Após a fase de implementação e a escrita do relatório, o melhor modelo obtido deve ser posto em produção. Desta forma, o grupo deverá implementar uma interface (*web* ou não) na linguagem de programação de sua preferência que mostre o modelo em funcionamento.

A interface construída deve ser apresentada na fase de Apresentações, podendo ser executada no computador de um dos integrantes do grupo.

As interfaces dos sistemas disponibilizados em `http://lasid.sor.ufscar.br/ml-tools/` podem ser utilizadas como base para ideias, porém será avaliado a criatividade e facilidade de uso da interface apresentada pelo grupo.

Apresentação

Cada grupo terá até 10 minutos para apresentar as análises realizadas e os resultados obtidos + 5 minutos para responder questionamentos.

Recomenda-se que cada grupo leve o seu próprio computador para evitar problemas de incompatibilidade de *software* e demoras na transição entre grupos.

CUIDADOS

Leia atentamente os itens a seguir.

1. O projeto deverá ser enviado pelo AVA observando as seguintes regras:
    - **Apenas um membro do grupo deverá submeter o projeto.**

- Submeter um arquivo `zip` chamado `GRUPOX_AM_PROJ.zip`, sendo que `X` corresponde ao número do grupo que fez a submissão. Ex: `GRUPO1_AM_PROJ.zip`. Tal arquivo deverá conter:
  - uma pasta chamada `GRUPOX_AM_PROJ` composta por duas subpastas: `IMPLEMENTACAO` (com o código-fonte completo, a base de dados usada e a interface), `RELATORIO` (todos os arquivos LaTeX e pdf) e um arquivo texto (`GRUPOX.txt`) com RA e nome de cada componente do grupo.

2. Não utilize acentos nos nomes de arquivos;
3. **Identificadores de variáveis:** escolha nomes apropriados;
4. **Documentação:** inclua cabeçalho, comentários e indentação no programa;
5. **Erros de compilação/execução:** nota **zero** no projeto para todos os membros;
6. **Tentativa de fraude:** cópia da Internet ou entre grupos implicará em nota **zero** na média para todos os membros do grupo.