Recuperação da Informação: Jogos

Alexandre Ferreira - afc2@cin.ufpe.br

Motivação

Dominios

- steam
- nuuvem
- americanas
- cultura
- fastshop
- kabum
- saraiva
- submarino
- walmart

Crawler: Heuristica

- Para alguns jogos, ocorre uma mudança na URL Quando se fala em páginas de jogos Como é o caso da steam
- Para esse tipo de site, eu coloquei prioridade quando aparece a regra ‘games’. Dessa forma, eu atribuo peso um para esses links zero para os demais
- Eu tive que rastrear as páginas específicas de jogos, e aumente a frequência destas 1000 páginas
- Eu acabei usando uma priorityqueue para priorizar as páginas de maior peso

Crawler: Possíveis melhorias

- Enquanto o sistema analisa a procura de links, ele procura também tags importantes como <div class = ‘classe de jogos’>
- Dessa maneira será possível prever para sites com crawler desenvolvidos não se restringindo apenas a url
- Um problema que aparece é o custo de analisar novamente em algumas situações.

As fases do Crawler

- Ele visita a página inicial de cada site com o basic_url e a partir dela extrai o Robots.txt
- Para usar Robots, foi necessário importar módulo
  - from urllib.robotparser import RobotFileParser
  - Sem esse módulo não haveria tanta precisão na hora de buscar as páginas permitidas pois os Robots não estão formatando em regex, mas foi permitido fazer sem ele
- Com a página inicial e os Robots, foi possível buscar em todas as âncoras por HREF que seja relevante
    - Dessa maneira evitou se a busca de páginas repetidas e de links fora do domínio como Facebook Twitter etc

- Após analisar a procura dos HREF, eu podia adicionar todos aqueles relevantes em uma priorityQueue da forma (-peso, URL)
    - (-peso) pois a PQ Implementada é uma min-heap

Crawler - Basic_crawler

- Considerada classe mestre de todos os crawlers, ela vai definir métodos básicos de busca e procura de páginas.
- Estes são alguns métodos abstratos que foram implementados para alguns dos crawler

O que o Crawler deve fazer

- Usar um classificador para dizer se é uma página relevante
- Estimar a melhoria do uso de um heurística nos crawlers
- Buscar mais domínios
- Ele deve melhorar a forma de estimar pesos para as páginas baixadas
    - A ideia de analisar em procura de elementos será o próximo passo
- Será necessário trabalhar na modularização do código
    - Formalizando as classes e os métodos abstratos
- Pensar em como armazenar as páginas a fim de não ocupar espaço

Classificador

Pré - processamento

- Remocao do HTML das paginas
- Remocao de pontos, virgulas, numeros e caracteres especiais
- Utilizcao de bag of words
  - Contagem e frequencia (stemming e stop wods)

Classificacao

- Sckit-learn
- Funcoes utilizadas:
  - Naives Bayes -> Gaussian NB
  - SVM -> SVC
  - Logistic Regression -> LogisticRegression
  - Decision Tree -> DecisionTreeClassifier
  - Multilayer Perceptron -> MLPClassifer
- Utilizacao de validacao cruzada estratificada

Classificação

- Calculo de desempenho
  - Acurarcia
  - Recall
  - Precisao
  - Tempo de Treinamento
- Utilizacao de 9 bag of words diferentes
  - contagem e frequencia
  - stemming, stop words, information gain

Information gain

- Utilizacao da funcao mutual_info_classif do scikit-learn
- Melhores palavras são as que falam dos requisitos do sistema

info_gain_tokens

Name	information_gain
5671	gb
9989	processor
5682	geforce
10701	requirements
6731	intel
10454	recommended
10281	radeon
515	amd
3678	directx
5900	graphics
10309	ram
5969	gtx
8848	nvidia
8233	minimum
10142	publisher
4186	electricity
13976	windows
6904	itch
8091	memory
7059	joinedfun

Desempenho por classificador

Naive Bayes

Naive_bayes

	Accuracy	Recall	Precision	Train time
tokenTfidf	0.55	1.0	0.5263157894736842	0.05501413345336914
tokenTfidf	0.7	0.4	1.0	0.05216789245605469
tokenTfidf	0.55	0.3	0.6	0.05243182182312012
tokenTfidf	0.6	0.2	1.0	0.05338788032531738
tokenTfidf	0.5	0.3	0.5	0.05365800857543945
tokenTfidf	0.6	0.3	0.75	0.05290699005126953
tokenTfidf	0.6	0.5	625	0.052935123443603516
tokenTfidf	0.6	0.5	625	0.051735877990722656
tokenTfidf	0.6	0.3	0.75	0.05172991752624512
tokenTfidf	0.6	1.0	0.5555555555555556	0.054270029067993164
stopwords	0.5	1.0	0.5	0.05924201011657715
stopwords	0.65	0.3	1.0	0.05944705009460449
stopwords	0.6	0.3	0.75	0.059567928314208984
stopwords	0.55	0.1	1.0	0.059098005294799805
stopwords	0.55	0.3	0.6	0.059417009353637695
stopwords	0.5	0.1	0.5	0.059046030044555664
stopwords	0.65	0.6	0.6666666666666666	0.059205055236816406
stopwords	0.7	0.6	0.75	0.0586240291595459
stopwords	0.55	0.1	1.0	0.059736013412475586
stopwords	0.6	1.0	0.5555555555555556	0.05931401252746582

SVM

svm

	Accuracy	Recall	Precision	Train time
tokenTfidf	0.35	0.0	0.0	0.5708308219909668
tokenTfidf	0.9	0.9	0.9	0.5695161819458008
tokenTfidf	0.55	0.6	0.5454545454545454	0.5786969661712646
tokenTfidf	0.9	1.0	0.8333333333333334	0.5715620517730713
tokenTfidf	0.55	0.8	0.5333333333333333	0.5702729225158691
tokenTfidf	0.55	1.0	0.5263157894736842	0.5718710422515869
tokenTfidf	0.55	1.0	0.5263157894736842	0.56461501121521
tokenTfidf	0.8	0.9	0.75	0.5708889961242676
tokenTfidf	0.85	0.7	1.0	0.5799951553344727
tokenTfidf	0.65	0.8	0.6153846153846154	0.5611429214477539
stopwords	0.5	0.0	0.0	0.5394940376281738
stopwords	0.9	0.8	1.0	0.5496160984039307
stopwords	0.55	0.1	1.0	0.535228967666626
stopwords	0.85	0.7	1.0	0.5513350963592529
stopwords	0.75	0.6	0.8571428571428571	0.5411360263824463
stopwords	0.8	0.6	1.0	0.5382959842681885
stopwords	0.55	0.1	1.0	0.5367438793182373
stopwords	0.6	0.2	1.0	0.5424880981445312
stopwords	0.55	0.1	1.0	0.5461809635162354
stopwords	0.9	1.0	0.8333333333333334	0.5381081104278564
stopwordsTfidf	0.35	0.0	0.0	0.5651431083679199

Logistic Regression

logistic_reg

	Accuracy	Recall	Precision	Train time
tokenTfidf	0.4	0.0	0.0	0.033470869064331055
tokenTfidf	0.9	0.9	0.9	0.0307769775390625
tokenTfidf	0.55	0.3	0.6	0.030128002166748047
tokenTfidf	0.9	0.8	1.0	0.030534029006958008
tokenTfidf	0.7	0.8	0.6666666666666666	0.03034496307373047
tokenTfidf	0.9	0.8	1.0	0.02997279167175293
tokenTfidf	0.75	0.9	0.6923076923076923	0.030852794647216797
tokenTfidf	0.95	0.9	1.0	0.030242919921875
tokenTfidf	0.7	0.4	1.0	0.030364036560058594
tokenTfidf	0.75	0.5	1.0	0.02938103675842285
stopwords	0.45	0.0	0.0	0.10046005249023438
stopwords	0.85	0.9	0.8181818181818182	0.0908501148223877
stopwords	0.45	0.3	0.42857142857142855	0.08510208129882812
stopwords	0.6	0.2	1.0	0.08189201354980469
stopwords	0.75	1.0	0.6666666666666666	0.08582520484924316
stopwords	0.8	0.9	0.75	0.08817100524902344
stopwords	0.55	1.0	0.5263157894736842	0.08296418190002441
stopwords	0.85	0.9	0.8181818181818182	0.09150505065917969
stopwords	0.65	0.4	0.8	0.08765101432800293
stopwords	0.9	1.0	0.8333333333333334	0.08200979232788086

Decision Tree

Decision_tree

	Accuracy	Recall	Precision	Train time
tokenTfidf	0.4	0.0	0.0	0.08304810523986816
tokenTfidf	0.65	0.4	0.8	0.07654619216918945
tokenTfidf	0.55	0.1	1.0	0.08194994926452637
tokenTfidf	1.0	1.0	1.0	0.09793806076049805
tokenTfidf	0.45	0.5	0.45454545454545453	0.07377099990844727
tokenTfidf	0.85	0.7	1.0	0.09335088729858398
tokenTfidf	0.75	0.8	0.7272727272727273	0.09162116050720215
tokenTfidf	0.85	0.8	0.8888888888888888	0.08868908882141113
tokenTfidf	0.6	0.2	1.0	0.09114289283752441
tokenTfidf	0.55	0.1	1.0	0.08179211616516113
stopwords	0.4	0.0	0.0	0.08458304405212402
stopwords	0.65	0.4	0.8	0.07678508758544922
stopwords	0.6	0.2	1.0	0.07715702056884766
stopwords	0.85	0.8	0.8888888888888888	0.08388304710388184
stopwords	0.5	0.6	0.5	0.06928610801696777
stopwords	0.85	0.7	1.0	0.09126806259155273
stopwords	0.9	1.0	0.8333333333333334	0.08618307113647461
stopwords	0.85	0.7	1.0	0.08011412620544434
stopwords	0.7	0.4	1.0	0.07685685157775879
stopwords	0.5	0.0	0.0	0.07753801345825195
stopwordsTfidf	0.4	0.0	0.0	0.08509302139282227

Multilayer Perceptron

mlp				
	Accuracy	**Recall**	**Precision**	**Train time**
tokenTfidf	0.9	1.0	0.8333333333333334	14.806880950927734
stopwords	0.95	0.9	1.0	5.671523809432983
stopwordsTfidf	0.9	0.9	0.9	26.54773998260498
stemming	0.95	0.9	1.0	6.54721999168396
stemmingTfidf	0.85	0.8	0.8888888888888888	17.51723885536194
stopNstem	0.95	0.9	1.0	8.818400859832764
info_gain	1.0	1.0	1.0	2.702031135559082
stopNstemTfidf	0.85	0.8	0.8888888888888888	16.256876945495605
token	0.95	0.9	1.0	7.531517028808594

O que é necessario para o futuro

- criar os crawler não feitos neste primeiro trabalho
- definir uma estrageia para definir se uma pagina é positiva ou negativa usando os 5 classificadores
- Sera possível salvar os vectorizes e os classificadores para não ter que rodar sempre?

FIM

FIM