UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
CURSO DE CIÊNCIA DA COMPUTAÇÃO

ANDREI POCHMANN KOENICH
PEDRO COMPANY BECK
TURMA A

RELATÓRIO - TRABALHO FINAL

Disciplina: Estruturas de Dados
Professora: Viviane Moreira

Porto Alegre, maio de 2021.

1 INTRODUÇÃO

O presente relatório tem por objetivo demonstrar dados e conclusões relacionadas ao uso da árvore ABP e da árvore AVL em dois programas relacionados com recuperação de informações com base em palavras-chave.

Os programas em questão foram desenvolvidos em linguagem C e simulam um buscador de palavras-chave. Consistem em uma aplicação dividida nos módulos de indexação e de consulta. A lógica de programação relacionada exclusivamente a esses módulos é equivalente nas duas aplicações, com diferença apenas na estrutura de dados que é utilizada (árvore ABP ou árvore AVL).

Na fase de indexação, ocorre a leitura de um arquivo texto de entrada, que é composto por vários textos, cada um correspondendo a um índice numérico (chamado de tweet). Nessa etapa, o programa armazena cada palavra de cada tweet do arquivo texto em um nó da estrutura de dados utilizada, assim como o número do tweet referente a essa palavra. Ao final do processo de indexação, cada nó da estrutura irá conter, portanto, uma palavra, além de uma lista simplesmente encadeada contendo todos os tweets que correspondem a essa palavra no arquivo texto de entrada. Mesmo que uma palavra ocorra mais de uma vez no mesmo tweet, a lista de tweets associada a essa palavra não possuirá números repetidos. Antes da inserção de cada palavra na estrutura, verifica-se se a palavra a ser inserida já não está alocada na árvore (ou seja, a palavra a ser inserida é comparada com todos os nós já existentes na estrutura), de forma a evitar a presença de palavras repetidas.

Durante toda a indexação, são computados o número de nós da árvore criada e a quantidade de comparações realizadas na árvore. Especificamente para a aplicação envolvendo árvore AVL, também é computada a quantidade de rotações executadas para realizar as inserções de todas as palavras em cada nó da árvore (para rotações simples e rotações duplas, incrementa-se o valor total da quantidade de rotações em uma e duas unidades, respectivamente). Tais informações serão impressas no arquivo texto de saída.

Na fase de consulta, ocorre a leitura de um arquivo texto criado especificamente para essa etapa. Tal arquivo contém as palavras a serem buscadas no arquivo texto de entrada que contém todos os tweets. Ocorrem, portanto, comparações envolvendo as palavras do arquivo texto de consulta com as palavras inseridas na árvore criada durante a fase de indexação, até que as palavra do arquivo de consulta sejam encontradas na estrutura. Caso uma palavra seja encontrada, ela é impressa no arquivo texto de saída junto com os tweets nos quais ela aparece. As palavras não encontradas durante a fase de consulta também são impressas no arquivo de saída, mas com a indicação de que a busca na estrutura não retornou resultados.

Durante a consulta, o número de comparações realizadas na estrutura de dados é computado e impresso no arquivo texto de saída.

Todos os testes realizados com o programa ocorreram mediante o uso de um arquivo texto de consulta contendo um total de vinte palavras aleatórias. Em relação aos arquivos textos de

entrada, utilizou-se um total de vinte arquivos, sendo dez arquivos usados para a geração de uma escala gráfica menor (contendo um número menor de tweets, iniciando com dez mil até atingir cem mil) e outros dez utilizados para a geração de uma escala gráfica maior (contendo um número maior de tweets, iniciando com cinquenta mil até atingir quinhentos mil), com cada escala correspondendo a uma tabela.

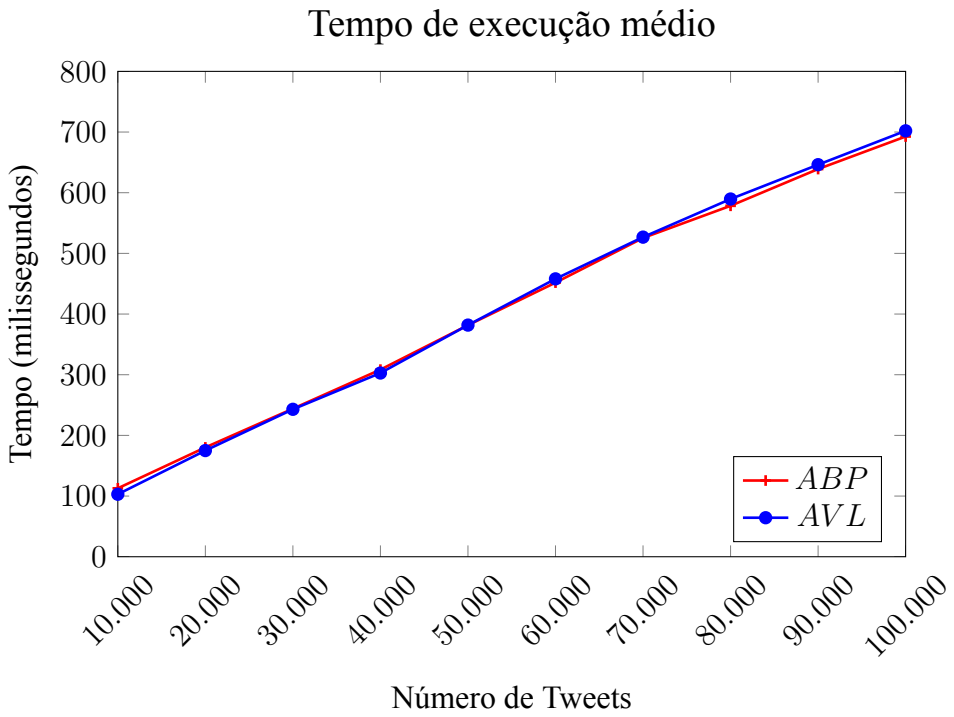
Nesse relatório, será analisada a relação entre o número de tweets recebidos pelo programa e, para cada estrutura, ao final da execução do programa: a quantidade de comparações realizadas (tanto na indexação quanto na consulta), o número de nós inseridos na estrutura durante a indexação e a altura da árvore. Especificamente para a aplicação envolvendo a árvore AVL, também será analisada a relação entre a quantidade tweets e a quantidade de rotações realizadas nas operações de inserção da árvore durante a indexação.

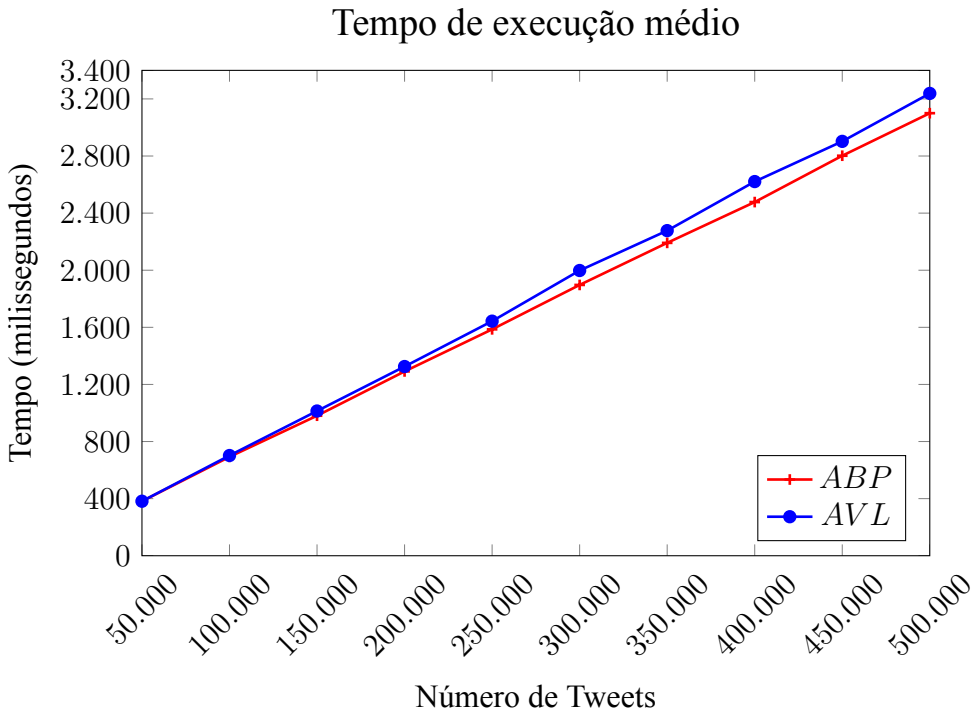
2 TEMPO DE EXECUÇÃO

O computador utilizado para determinar o tempo de execução de cada programa, mediante uma variação do número de tweets presentes no arquivo texto de entrada, utiliza um processador intel core i5-7400 @ 3.00 GHz. Considerando que podem ocorrer variações no tempo de processamento (com uma mesma estrutura e um mesmo número de tweets), para cada quantidade de tweets específica, realizou-se a execução cinco vezes, e um cálculo da média aritmética com os tempos de processamento obtidos em cada um desses cinco testes, conforme exposto nas tabelas abaixo.

A análise do tempo de execução revela um crescimento linear. Nota-se que, na maioria dos casos, o programa que faz uso da árvore AVL necessita de mais tempo para ser processado (considerando um mesmo número de tweets) em relação ao programa que faz uso da árvore ABP como estrutura, conforme evidenciado pela diferença entre os tempos de processamento para cada estrutura, evidenciado nas tabelas abaixo.

A predominância do tempo de execução envolvendo a árvore AVL sobre o tempo de execução envolvendo árvore ABP é justificada pelas operações de rotação (simples ou dupla) que são necessárias quando da inserção de novas palavras na estrutura, proporcionando um custo computacional maior, que é inexistente nas operações envolvendo a árvore ABP.





Número de Tweets	Tempo de execução médio (ms)		
	ABP	AVL	Diferença (ABP-AVL)
10000	112,8	103,0	9,8
20000	180,2	175,0	5,2
30000	244,0	243,0	1,0
40000	308,4	302,8	5,6
50000	381,6	381,8	-0,2
60000	451,8	458,0	-6,2
70000	525,4	526,8	-1,4
80000	578,4	589,6	-11,2
90000	639,0	646,2	-7,2
100000	692,6	702,0	-9,4

Número de Tweets	Tempo de execução médio (ms)		
	ABP	AVL	Diferença (ABP-AVL)
50000	381,6	381,8	-0,2
100000	692,6	702,0	-9,4
150000	980,8	1013,8	-33,0
200000	1291,0	1325,4	-34,4
250000	1585,8	1643,2	-57,4
300000	1897,0	1998,0	-101,0
350000	2191,6	2277,4	-85,8
400000	2477,4	2621,0	-143,6
450000	2802,0	2902,6	-100,6
500000	3100,0	3238,4	-138,4

Tabela: Tempo de execução ABP

Número de Tweets	Tempo de Execução (ms)					
	1	2	3	4	5	Média
10000	112	127	108	106	111	112,8
20000	188	181	174	179	179	180,2
30000	242	246	249	242	241	244,0
40000	311	313	290	316	312	308,4
50000	388	377	374	385	384	381,6
60000	448	442	458	462	449	451,8
70000	530	528	525	524	520	525,4
80000	579	578	585	563	587	578,4
90000	645	648	637	630	635	639,0
100000	699	703	694	681	686	692,6
50000	388	377	374	385	384	381,6
100000	699	703	694	681	686	692,6
150000	1000	973	990	962	979	980,8
200000	1287	1286	1283	1294	1305	1291,0
250000	1578	1592	1581	1582	1596	1585,8
300000	1906	1898	1889	1895	1897	1897,0
350000	2204	2182	2192	2177	2203	2191,6
400000	2474	2469	2470	2477	2497	2477,4
450000	2788	2791	2779	2833	2819	2802,0
500000	3099	3094	3089	3096	3122	3100,0

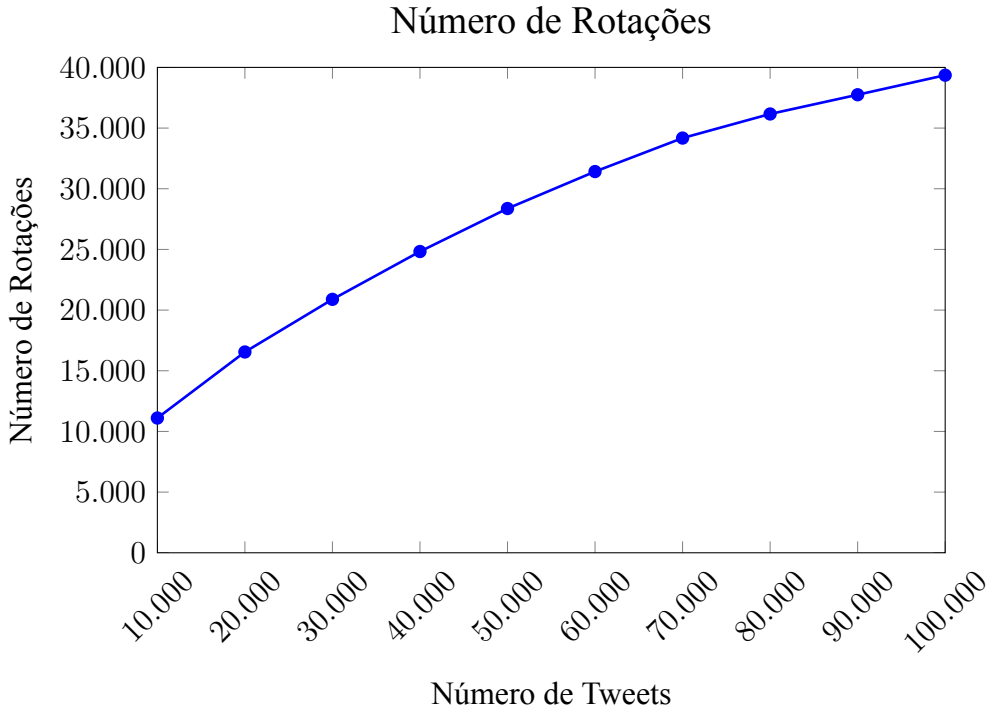
Tabela: Tempo de execução AVL

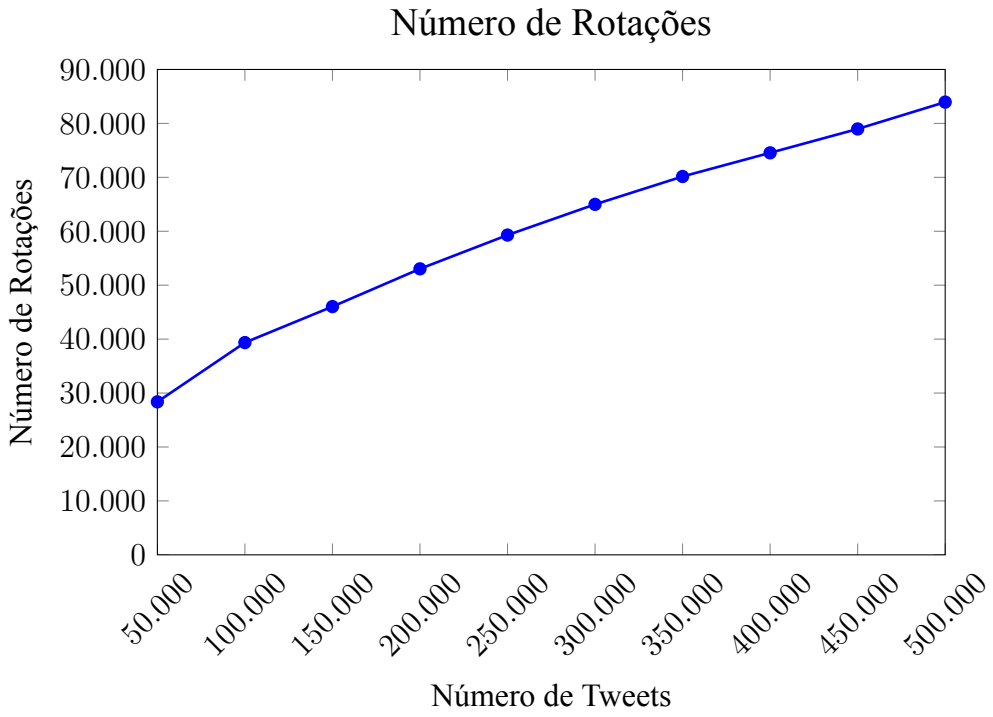
Número de Tweets	Tempo de Execução (ms)					
	1	2	3	4	5	Média
10000	107	98	100	108	102	103,0
20000	189	177	164	177	168	175,0
30000	258	247	245	219	246	243,0
40000	321	290	307	312	284	302,8
50000	385	385	371	390	378	381,8
60000	453	457	460	461	457	458,0
70000	530	527	529	523	525	526,8
80000	598	594	586	583	587	589,6
90000	664	649	631	647	640	646,2
100000	709	669	706	709	717	702,0
50000	385	385	371	390	378	381,8
100000	709	669	706	709	717	702,0
150000	1033	1008	1001	1005	1022	1013,8
200000	1332	1307	1330	1332	1326	1325,4
250000	1651	1629	1656	1636	1644	1643,2
300000	2126	1973	1956	1949	1986	1998,0
350000	2284	2280	2270	2283	2270	2277,4
400000	2741	2586	2575	2589	2614	2621,0
450000	2895	2898	2910	2900	2910	2902,6
500000	3224	3223	3231	3245	3269	3238,4

3 NÚMERO DE ROTAÇÕES

A análise do número de rotações revela um crescimento semelhante ao logarítmico, e envolve somente o programa que faz uso da árvore AVL como estrutura principal, posto que as inserções em árvores ABP não demandam rotações.

Nota-se que o crescimento do número de rotações é desacelerado ao aumentar o número de tweets, o que é demonstrado na coluna da tabela correspondente à diferença entre o número de rotações de cada intervalo: aumentando o número de palavras, essa diferença decresce cada vez mais. Isso é motivado pelo fato de que, quanto mais tweets já foram analisados, mais palavras diferentes já foram inseridas na estrutura, de modo que as indexações passam a exigir cada vez menos a utilização de rotações, uma vez que o programa não permite inserções de palavras repetidas na estrutura.



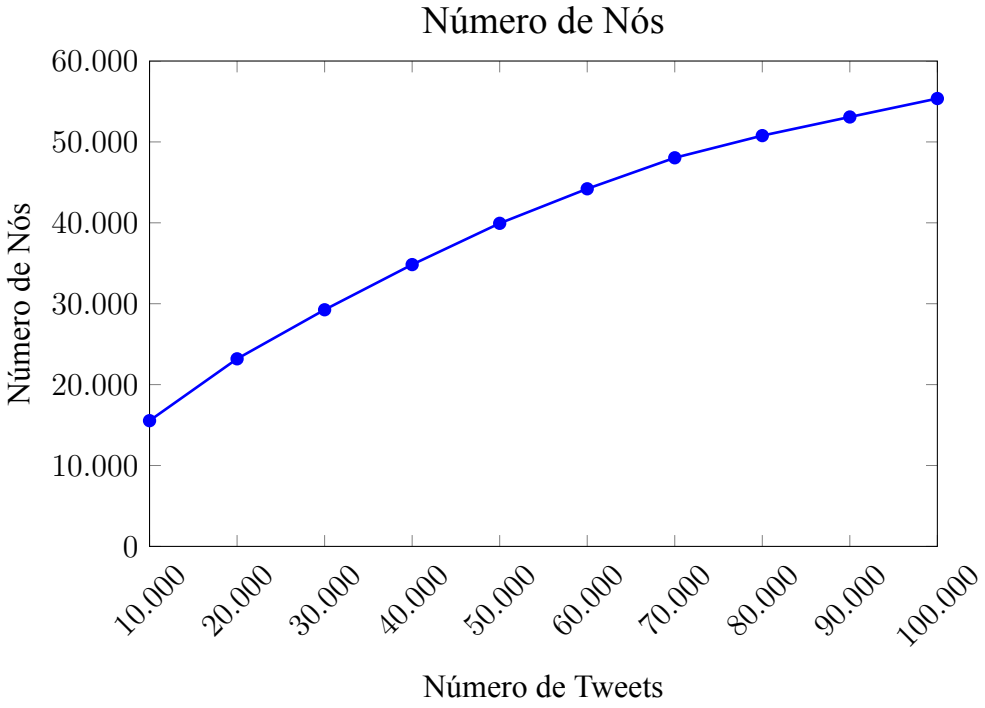


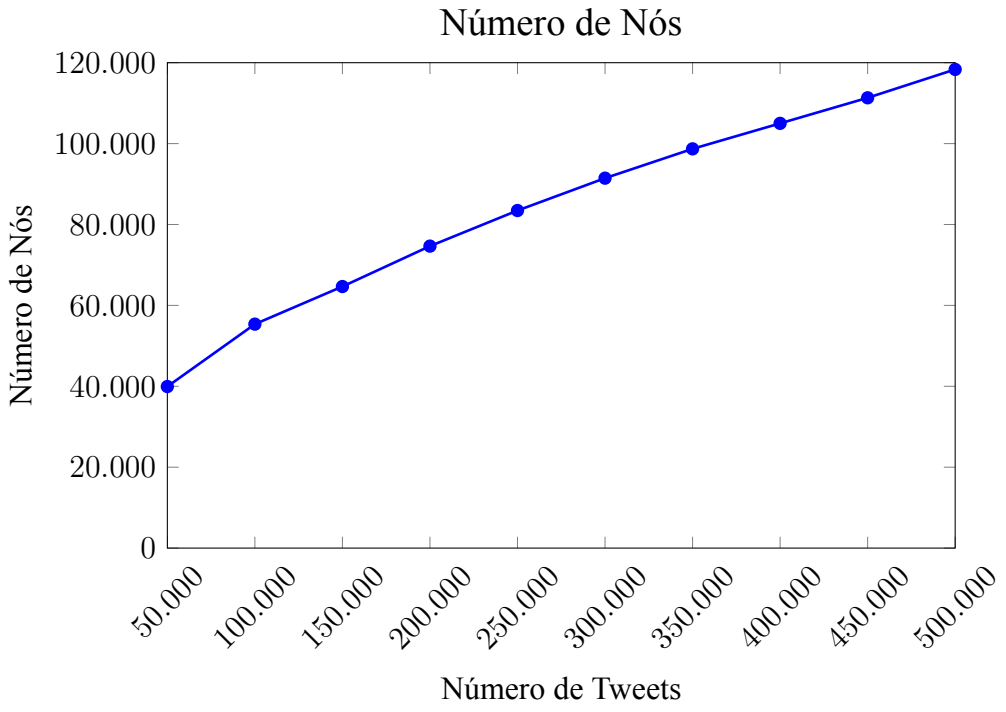
Número de Tweets	Número de Rotações
10000	11105
20000	16548
30000	20885
40000	24827
50000	28370
60000	31417
70000	34177
80000	36162
90000	37747
100000	39358

Número de Tweets	Número de Rotações
50000	28370
100000	39358
150000	46030
200000	53035
250000	59293
300000	64986
350000	70154
400000	74543
450000	78969
500000	83950

4 NÚMERO DE NÓS

A análise do número de nós revela um crescimento semelhante ao logarítmico, e é único tanto para a aplicação envolvendo a árvore AVL quanto para a que envolve a árvore ABP, pois os arquivos de entrada são os mesmos. Assim como ocorreu com o gráfico envolvendo as operações de rotação em árvore AVL, nota-se uma desaceleração do crescimento, à medida que o número total de tweets aumenta. Novamente, isso é justificado pelo fato de que um número cada vez menor de palavras novas é inserido conforme o número de tweets é aumentado, pois a árvore contém cada vez mais palavras a cada execução, e não são permitidas inserções de palavras repetidas na estrutura.

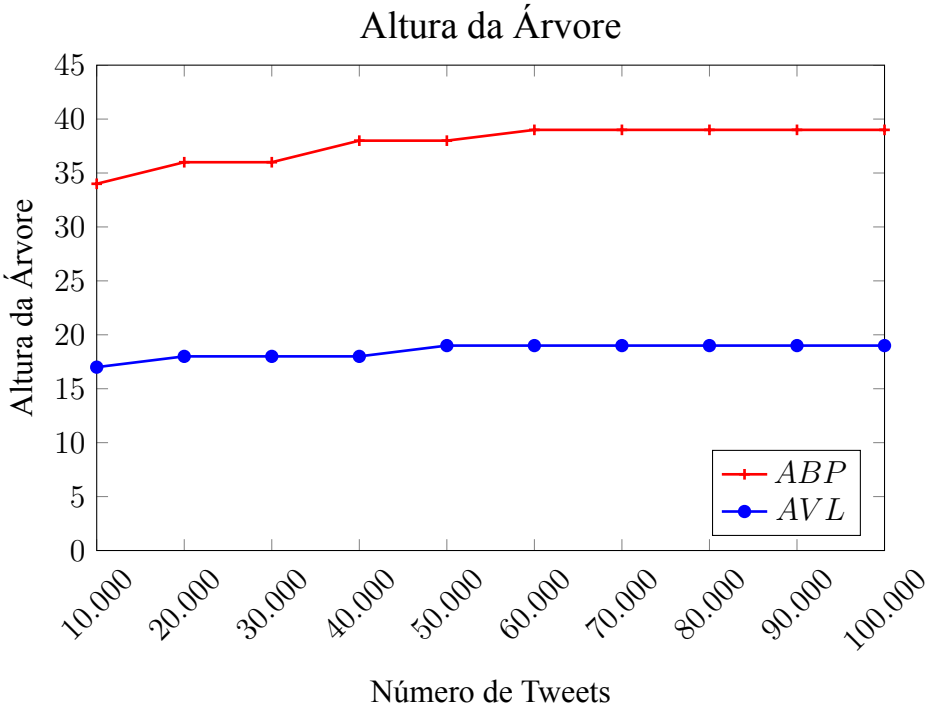


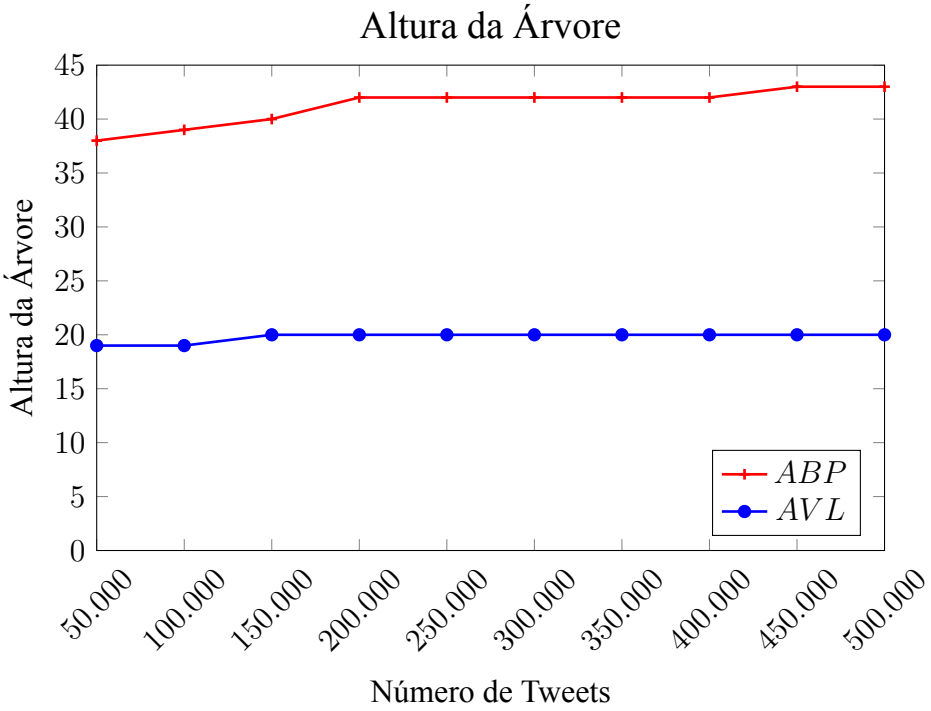


Número de Tweets	Número de Nós	Número de Tweets	Número de Nós
50000	39938	10000	15542
100000	55363	20000	23188
150000	64661	30000	29252
200000	74654	40000	34835
250000	83453	50000	39938
300000	91454	60000	44206
350000	98700	70000	48039
400000	105001	80000	50779
450000	111321	90000	53083
500000	118341	100000	55363

5 ALTURA DA ÁRVORE

A análise da altura de cada árvore revela uma diferença significativa entre as duas estruturas em relação a essa característica, tendo em vista que as operações específicas relacionadas às inserções de nós na árvore AVL garantem que a árvore tenha uma altura menor em comparação com a árvore ABP. Nota-se que no ponto de partida dos gráficos (correspondente aos arquivos de entrada contendo dez mil e cem mil tweets), já existe uma diferença entre as alturas. Conforme o esperado, a altura das árvores aumenta conforme o número de tweets sofre um incremento, pois novas palavras são inseridas na estrutura, o que demanda uma expansão na altura das árvores.





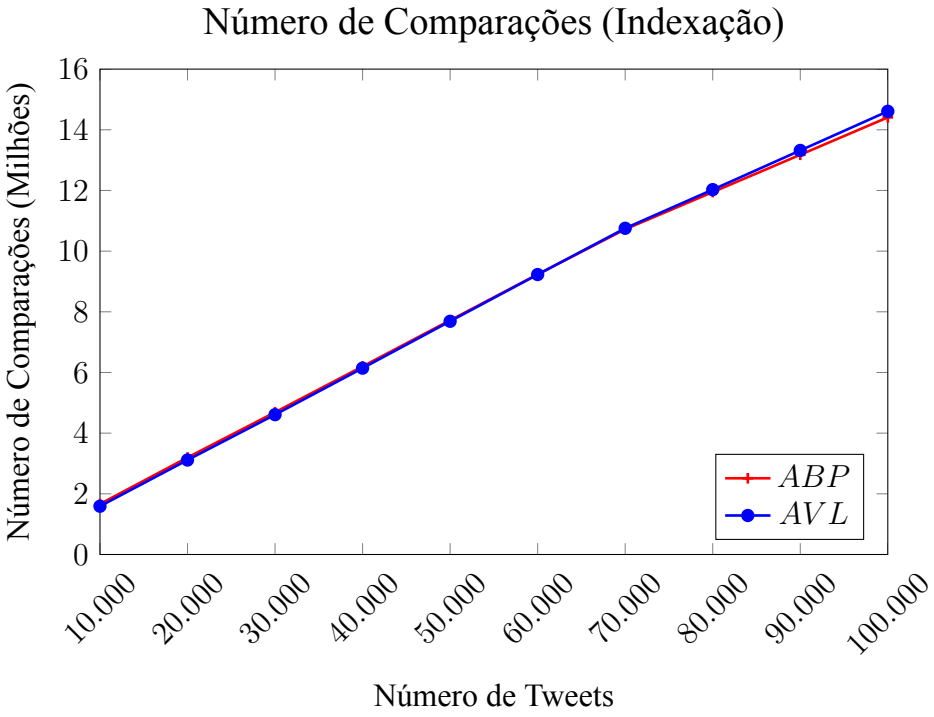
Número de Tweets	Altura da Árvore		
	ABP	AVL	Diferença (ABP-AVL)
10000	34	17	17
20000	36	18	18
30000	36	18	18
40000	38	18	20
50000	38	19	19
60000	39	19	20
70000	39	19	20
80000	39	19	20
90000	39	19	20
100000	39	19	20

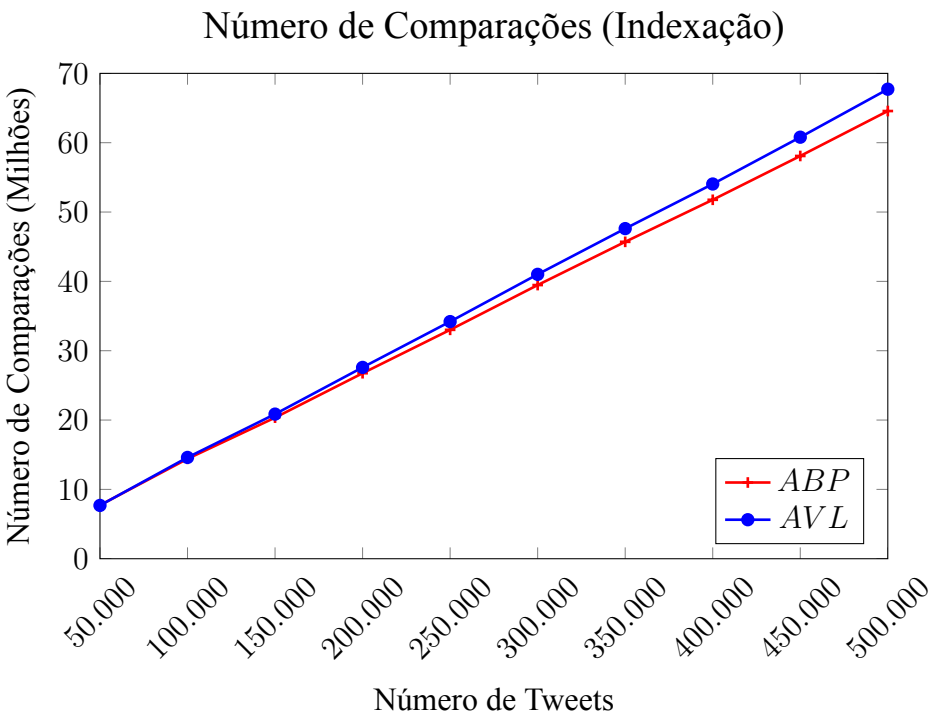
Número de Tweets	Altura da Árvore		
	ABP	AVL	Diferença (ABP-AVL)
50000	38	19	19
100000	39	19	20
150000	40	20	20
200000	42	20	22
250000	42	20	22
300000	42	20	22
350000	42	20	22
400000	42	20	22
450000	43	20	23
500000	43	20	23

6 NÚMERO DE COMPARAÇÕES

6.1 Indexação

A análise do número de comparações realizadas na operação de indexação revela um crescimento semelhante ao linear. Nota-se que, nos momentos iniciais do gráfico com a escala menor, o número de comparações realizadas na árvore AVL é menor, e a situação se inverte quando o número de tweets analisados é próximo de setenta mil. Após isso, o número de comparações realizadas na árvore ABP é sempre menor, o que demonstra que a eficiência gerada pelas operações de rotação na árvore AVL (em comparação com a quantidade de comparações na árvore ABP) ocorre somente até um determinado ponto.



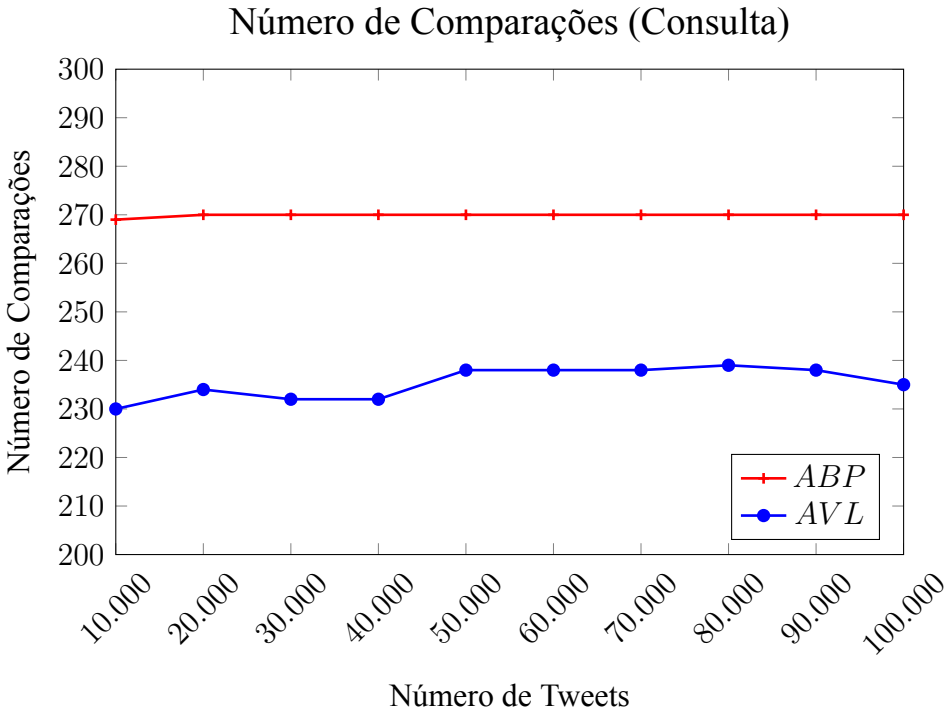


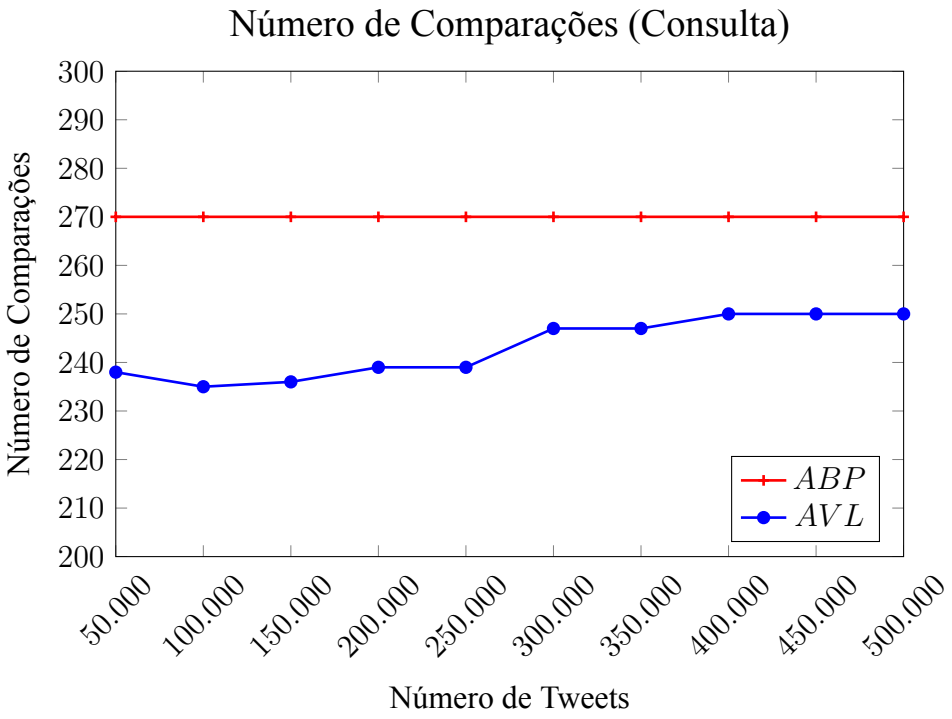
Número de Tweets	Número de Comparações		
	ABP	AVL	Diferença (ABP-AVL)
10000	1663916	1591355	72561
20000	3196805	3114320	82485
30000	4687880	4609710	72170
40000	6198785	6142915	55870
50000	7719435	7687769	31666
60000	9235018	9228278	6470
70000	10720561	10753676	-33115
80000	11946255	12027262	-81007
90000	13176357	13318962	-142605
100000	14409181	14608905	-199724

Número de Tweets	Número de Comparações		
	ABP	AVL	Diferença (ABP-AVL)
50000	7719435	7687769	31666
100000	14409181	14608905	-199724
150000	20348192	20857848	-509656
200000	26751716	27593013	-841297
250000	33011356	34200372	-1189016
300000	39468763	41014856	-1546093
350000	45703117	47611118	-1908001
400000	51766825	54042812	-2275987
450000	58085420	60793470	-2708050
500000	64567578	67714928	-3147350

6.2 Consulta

A análise do número de comparações realizadas na operação de consulta revela, na aplicação envolvendo árvore AVL, um crescimento pouco significativo à medida que o número de tweets é ampliado, enquanto que na aplicação envolvendo árvore ABP não há crescimento. Nesse módulo em específico, é possível perceber a eficiência gerada pelas operações de rotação na árvore AVL, que desta vez é predominante em todos os intervalos numéricos de tweets considerados.





Número de Tweets	Número de Comparações		
	ABP	AVL	Diferença (ABP-AVL)
10000	269	230	39
20000	270	234	36
30000	270	232	38
40000	270	232	38
50000	270	238	32
60000	270	238	32
70000	270	238	32
80000	270	239	31
90000	270	238	32
100000	270	235	35

Número de Tweets	Número de Comparações		
	ABP	AVL	Diferença (ABP-AVL)
50000	270	238	32
100000	270	235	35
150000	270	236	34
200000	270	239	31
250000	270	239	31
300000	270	247	23
350000	270	247	23
400000	270	250	20
450000	270	250	20
500000	270	250	20

7 CONCLUSÃO

Com base nas análises realizadas nesse documento, é possível concluir que há alguns benefícios práticos de se usar uma árvore AVL, no lugar de árvores ABP.

O maior benefício de árvores AVL se dá na parte de consulta. Foram necessárias em média 13% menos consultas na menor escala, e 10% menos consultas na maior escala. Então caso já se tenha uma árvore AVL pronta, um programa que apenas a consulta é mais rápido porque realiza menos comparações, porém como esse programa realiza primeiro a indexação e depois a consulta, os tempos de execução foram bastante similares.

Entretanto, na maioria dos casos e, principalmente em casos com maior número de tweets, o tempo de execução médio foi mais lento com árvores AVL, devido às rotações na indexação. Além disso, na parte de indexação, para bases maiores, foram necessárias mais comparações em árvores AVL se comparado com árvores ABP.