Frequent Itemset Mining for Query Expansion in Microblog Ad-hoc Search

Younos Aboulnaga   Charles L. A. Clarke

David R. Cheriton School of Computer Science, University of Waterloo
200 University Avenue, Waterloo, ON, Canada N2L 3G1
{yaboulna,claclark}@uwaterloo.ca

ABSTRACT

The high volume of Tweets arriving every second and the requirement to index them in real time emphasize the importance of the computational complexity of algorithms used to process them. In this paper, we investigate the use of Frequent Itemsets Mining to quickly discover patterns that can later be used for query expansion. Frequent Itemsets Mining (FIM) has been highly adopted to mine data streams because of its computational simplicity and the possibility to parallelize some of its steps. Initial experiments using the TREC 2011 Microblogs track queries showed that it is possible to improve the performance of BM25, however this was not the case with the 2012 queries. Our analysis of the difference in performance provides insight about how to make best use of FIM for microblog search.

1. INTRODUCTION

The short length of Tweets is a challenge to many of the algorithms developed for searching other corpora, but it also makes it possible to apply other algorithms developed for other forms of data. The computational simplicity of Frequent Itemsets Mining (FIM) makes it an attractive option in situations where the processing time is important, such as real-time ad-hoc search. It is also straightforward to model Tweets as Itemsets, since the mean number of terms in a Tweet is small. In the TREC 2012 Microblogs track's dataset, Tweets written in Latin characters contained 9.637 terms on average. Besides the algorithmic convenience, itemsets also fit the Bag of Words model which neglects the order of terms. In this paper we use itemsets mined from the whole dataset to expand queries, and experiment with different techniques for selecting expansion terms.

The rest of the paper is organized as follows. Initially we describe the indexing process in section 2, and we explain our choice of baseline in section 3. Then we cover the Frequent Itemsets Mining process in section 4. In section 5 we give the details of how we use the mined itemsets for query expansion, and in section 6 we report the performance of our methods on the TREC topics. Finally, we conclude by section 7 where we also highlight our future work.

2. INDEX

We use Lucene 3.6 [1] as an inverted index. Our tokenizer keeps only Latin characters (Unicode code points less than 'u024F'), numbers, and the characters {'@', '#', '_'}. All URLs are replaced by the token 'URL', and runs of the same character is reduced to only 3 repetitions (for example, 'coooool' is replaced by 'coool'). Hashtags are stored twice, with and without the the '#' character. All tokens are stored both in their original form, and after applying Lucene's implementation of the Porter stemmer. We use the original form to select the subset of documents to score, and the stemmed form while scoring. This avoids the disadvantage that stemming retrieves irrelevant documents, while taking advantage of it to count words about the same concept towards the same term frequency. While words sharing the same stem can mean totally different concepts in the context of different documents, it is very likely that they have the same meaning within one document, and this meaning is probably the one desired if the document is retrieved using the unstemmed query term.

3. BASELINE

The baseline on which we improve uses the BM25 ranking function used in [2]:

$$\begin{aligned} BM25_d &= \textstyle\sum_t q_t.idf_t, \\ idf_t &= log(\tfrac{N-df_t+0.5}{df_t+0.5}) \end{aligned} \tag{1}$$

This is analogous to the Okapi BM25 equation with k_1 set to 0 to annihilate the effect of term frequency, so a Tweet with one occurrence of a query term is as good as one with many. The short length of Tweets restricts the term frequency to numbers close to unity, and there is no reason to believe that a Tweet with two or three occurrences is more relevant than a Tweet with one occurrence. Actually Tweets with many occurrences are likely to be spam, or totally uninformative. Moreover, setting k_1 to 0 also annihilates the effect of document length normalization, favouring long Tweets that make most use of the allowed length. It is noteworthy that we had arrived at this formula by looking for the best values for k_1 and b using a grid search. We varied the values of b from 0 to 1 in increments of 0.1 and the value of k_1 from 0 to 100, and tested the performance of Okapi BM25 on the TREC 2011 Microblog track qrels. There are

[1] http://lucene.apache.org/

other values that performed nearly as well or even slightly better, but the intuitive explanation of the selected values make them less likely to be over fitting the data.

4. FREQUENT ITEMSETS MINING

The FIM algorithm we use is a slightly modified version of Mahout's[2] implementation of PFP [5], a Map/Reduce adaptation of the FP-Growth algorithm [3]. The FP-Growth algorithm is scalable to a large number of vocabulary terms (items), because it skips the candidate generation step which is the bottle neck of Apriori [1] like algorithms. By taking advantage of the parallelization capabilities of Map/Reduce, it is possible to speed up the mining process almost arbitrarily because the overhead of adding more nodes is subtle, as reported in [5]. In our experiments we used only one machine, a quad-core AMD Phenom PC with 8 GB of RAM. Even on such modest machine, we could always get the results of mining a window of any length in less time than the window length. The reported runs do not respect the real time requirement because the Frequent Itemsets used for expansion are mined from the whole dataset. To simulate using the itemsets available at a certain point of time we first need to adapt the mining algorithm for a streaming environment. Parallel FIM algorithms that work on streams are mostly Apriori like because the FP-Tree structure is hard to update, but this is not our main area of interest.

The PFP algorithm avoids running out of memory by keeping only a certain amount of itemsets for each item, selecting the ones with the highest support. We modify it by using measures that go beyond merely counting and take into account the semantic relations between terms. The results reported use itemsets mined with support as the measure in intermediate stages, but finally when selecting which itemsets to keep for each term we use the Normalized Mutual Information (NMI) of the whole itemset I with the *head* term h:

$$NMI(I,h) = \frac{\sum_{t \in I} p(t,h) \ln \frac{p(t,h)}{p(t)p(h)}}{-\sum_{t \in I} p(t,h) \ln p(t,h)} \qquad (2)$$

The probabilities required for calculating NMI are easily estimated from the FP-Tree. The FP-Tree is a prefix tree in which the count at each node represent the number of times the term in this node occurs in itemsets along with items in higher levels. The structure also keeps links between nodes of the same term. Therefore to estimate the joint probability of two terms it is easy to get all paths between them, then sum the counts at lower level nodes. This is an estimate not an exact calculation because subtrees for which the root's count is less than the support are pruned. The tree is also kept within a certain size limit by increasing the support kept in the tree every time it reaches the size limit. If support is used to select the final set of itemsets then this increase of support in intermediate stages only prunes itemsets that wouldn't be selected for the final result anyway, according to [5]. To avoid ending up with patterns including only terms that have very high support, such as stop words, we remove the top 1 percentile of terms.

Mutual Information is a better measure than support for the final selection since it is a measure of association. Frequent Itemsets Mining is usually a preliminary stage for associate rules mining which uses many measures other than support, such as confidence, lift and Pearson's correlation to name a few. We chose Normalized Mutual Information because of its suitability for use with natural languages since it is robust against low frequency data. It is also a measure whose value is bounded even for itemsets with different lengths. In our work in progress we are empirically evaluating the properties of different other measures.

5. QUERY EXPANSION

To use the mined frequent itemsets for query expansion, they are first indexed just like Tweets. The itemsets corpus is searched with the original query, using BM25 for ranking. A few terms are selected from the itemsets result set, using one of a variety of methods. In later subsections, we describe the two best performing expansion term selection techniques, as well as a "control" Pseudo Relevance Feedback technique. The number of expansion terms that worked best with the TREC 2011 qrels is 10 expansion terms for each query term. The weight of the expansion terms are set so that their total weight is equal to the total weight of the original query, thus reducing the effect of concept drift.

5.1 Terms from most relevant itemsets

Expansion terms are added from patterns with the highest BM25 scores, until the required number of distinct expansion terms are added. This is the most straightforward method yet it is the most effective one. Other methods were attempts to improve on this simple BM25 relevance ranking, but none of them could achieve better performance. By using BM25 for retrieving relevant itemsets we balance between adding expansion terms that increase diversity and ones that increase specificity of the query. More diversity is achieved by using itemsets that satisfy the disjunction of the original query, but this causes severe concept drift. On the other hand, using a conjunction of the original query leads to filtering the results to very specific Tweets - the ones from which the itemsets were mined - leading to a result set with at least *minimum support* repetitions of very similar Tweets. The IDF weighting of terms in BM25 strikes a good balance for selecting itemsets, just like it does for Tweets.

5.2 Terms from clusters of itemsets

We attempted a variety of methods to select expansion terms by using a matrix of itemset to term features, where each row represents an itemset and each column represents a term. Instead of using binary indicators of presence of a term in an itemset, we use a value similar to the one used in [8]. The value in cells of a column is the change in the IDF of the column's term when calculated on the corpus of relevant itemsets from that calculated on the whole corpus of Tweets. The value is also scaled by the smoothed probability of the term in the Tweets corpus. This value gives more weight to terms whose IDF has *decreased* in the relevant itemsets corpus compared to that in the Tweets corpus, but scales this weight by the term's probability. This value is actually the negative of the KL-divergence between the probability of the term in the Tweets corpus and the relevant itemsets corpus. The performance of several algorithms improved by using this elaborate value. It is given by the following equation:

[2] http://mahout.apache.org/

$$-KLD = (\ln \frac{N_c}{df_{t,c}} - \ln \frac{N_r}{df_{t,r}}) * p_{t,c} \tag{3}$$

This itemset to terms matrix was used as an input to many algorithms for selecting the *important* terms. We attempted Singular Value Decomposition (SVD) as in [8], Markov chain word translation model [4], PageRank and clustering. Terms chosen by clustering itemsets caused the biggest improvement in performance, and it is also the easiest method to explain. It simply tries to expand the query by terms from different possible concepts which are assumed to be represented by clusters of itemsets in the relevant result set.

We use the XMeans [6] implementation in Weka[3] to cluster itemsets according to the matrix given above. The matrix is standardized to avoid any unexpected behaviour because of great variation in the values in the matrix. Then expansion terms are selected by taking the term closest to each cluster centroid in turn, then the second closest and so on. The distance of a term from a cluster centroid is the mean of the distances of itemsets in which it is a member.

5.3 Terms for clusters of Tweets

This is the same as expansion using terms from clusters of itemsets, but using Tweets relevant to the original query instead of itemsets; that is, using Pseudo Relevance Feedback (PRF). This is meant as a "control" run to assess whether itemsets mining is useful. Better performance is sometimes achieved using PRF, however the expansion terms from the itemsets makes more sense than those from the relevant Tweets. Tables 1 and 2 show the expansion terms of the first 5 topics using both methods. The topics' queries are given in appendix A. Terms from the original query are marked by an asterisk in the expanded query.

6. EVALUATION

Table 3 shows Precision at 30 (P@30), Mean Average Precision (MAP) and Reciprocal Precision (R-Prec) for the base line (BM25) and for the different expansion methods, nFromTop described in section 5.1, clusterFIS described in section 5.2 and clusterTwts described in section 5.3. Performance is reported on both the TREC 2011 and 2012 topics, using all relevant qrels for both sets of topics, as well as only highly relevant qrels for the 2012 topics. The effect of query expansion on the 2011 topics is positive, but it has virtually no effect on the 2012 topics. Out of the 60 topics of 2012, only topic no. 84 got improved and topics no. 87, 90 and 99 got worse. However on the 49 topic of 2011, our baseline is in the highest 10 percentile and the query expansion beats the best run reported in 2011. Following, we explore the possible causes for such a different performance.

One reason could be that the expansion with frequent itemsets work well on certain types of queries, like those which include a named entity for example. We performed an Analysis of Variance (ANOVA) of the performance across different categories of topics, based on the three categorizations in [7]. One-way ANOVA tests of the three categorizations don't show a significant variance of performance across different categories, neither do two- and three-way tests. Even though the categories are unbalanced, the normality of residuals was verified by QQ-plots and the homogeneity of variance was verified by the Levene test.

Topic	Expanded Query
MB051	@britishexpat, government*, #fed, debt, #expats, cuts*, budgets, #government, fed, british*, @sion_israel, #british, expats, #debt, #cuts
MB052	outbreak, probes, bedbug*, contribute, @i_kill_termites, abating, upends, salts, h1n1, infestation, swine, @healthqd, paralysis, obesity, heart, epidemic*, #obesity, @n1hc, specialists, proportions, willpower, #heart
MB053	superclasico, rcl, #condo, river*, riverboat, viking, web5, web3, web1, #superclasico, #michigan, #fishing, #nature, overlooking, superclĞsico, fishing, #boat, news1, #park, plunges, athens, #lincoln, msc, itineraries, @kathsellshomes, condo, cruises*, @cruiseschedules, lincoln, getaways, boat*, park, nature, nile, @rcaribbeancruis
MB054	19.99, oakley, 27.99, twitenna, of, @imalwayzsmacked, preisvergleich, #the, #2chmatome, #preisvergleich, steepandcheap, cigarmonster, the*, #and, #premier, @soccerts, @trends_internet, 24.99, 2chmatome, #of, and, daily*, hooded, robusto, #twitenna, 39.99
MB055	shark, shari, dealcenter, blues, @andrewpain1974, with, maqui, ezinearticles4u, #you, watchers, of, captin, @rayhattersley, #rhythm, almond, #lift, #white, rhythm, #blues, #science, loss*, you, #eating, #beauty, ambria, #with, weight*, diets, white, @puwisdom, science, #and, @pulistbook, lift, dingle, @miracleweight, @tweettraffic4u, #diets, oranges, #of, @aase25, and*, berries*, #no, navigator, @pubooks, eating, beauty, @music_mattters

Table 1: Expansion using Frequent Itemsets

Another reason could be that the weights given to the expansion terms were too low for them to have any effect. However, increasing the weights in further experiments, with the same setting of the official runs, caused severe concept drift. This leads us to believe that the actual reason is that the expansion terms are not really relevant to the query. Part of the reason behind this might be the way users employ hashtags, because most of the expansion terms are actually hashtags. The problem with hashtags is the versatility of their use across different topics and even languages. For example, the hashtag "#fishing" is used in many CKJ Tweets, and the hashtags "#obesity" and "#fit" are used in many Tweets not pertaining to the topic query. This can be overcome by selecting expansion terms that have stronger association with query.

7. CONCLUSION AND FUTURE WORK

We have described our use of Frequent Itemsets for query expansion in the TREC 2012 microblog ad-hoc search task. We have seen promising results using the TREC 2011 topics, but the expansion had no effect on the TREC 2012 topics. We believe that the approach is promising and intend to improve it by using different measures for selecting the expansion terms from the mined Frequent Itemsets, and by mining Itemsets that yield strong association rules.

[3] http://www.cs.waikato.ac.nz/ml/weka/

Method	2012 High Rel					2012 All Rel			2011 All Rel		
	P@30	MAP	R-Prec	N > median	N ≥ median	P@30	MAP	R-Prec	P@30	MAP	R-Prec
BM25	0.1944	0.1623	0.1926	20	43	0.3814	0.2442	0.2982	0.4177	0.3511	0.3968
nFromTop	0.1949	0.1657	0.1922	20	43	0.3814	0.2490	0.3010	0.4525	0.3764	0.4130
clusterFIS	0.1955	0.1646	0.1930	20	43	0.3819	0.2467	0.2991	0.4204	0.3501	0.3925
clusterTwt	0.1831	0.1620	0.1824	17	41	0.3650	0.2454	0.2933	0.4381	0.3596	0.4029

Table 3: Performance on the 2012 and 2011 topics

Topic	Expanded Query
MB051	cut, government*, you, fannie, cameron, on, of, stake, tax, for, down, eat, are, cuts*, out, british*, cutting, the, about, seek, s, to, i, just, a, be, spending, from, and, it, is, in, if
MB052	of, bedbug*, that, health, its, are, my, this, the, t, by, to, i, new, a, spreading, from, an, and, it, is, in, epidemic*, all
MB053	river*, with, travel, go, on, of, that, for, down, cruise, you, results, want, take, but, best, the, about, s, m, to, i, new, a, cruising, at, #cruise, and, cruises*, is, in, boat*, get
MB054	your, go, of, stories, are, my, you, out, via, the*, every, today, to, i, a, gets, from, top, really, and, it, is, daily*
MB055	benefits, with, acai, their, diet, on, of, wt, health, plans, ber, for, first, how, are, loss*, you, this, mt, lose, weight*, best, the, include, t, s, to, behind, new, a, precisely, really, workouts, and*, facts, berry, berries*, it, is, in, has, juice, will, what, effective, healthy, pills

Table 2: Expansion using Pseudo Relevance Feedback

APPENDIX

A. TOPICS' QUERIES

MB051 British Government cuts

MB052 Bedbug epidemic

MB053 river boat cruises

MB054 The Daily

MB055 berries and weight loss

B. REFERENCES

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc.

[2] P. Ferguson, N. OHare, J. Lanagan, A. Smeaton, O. Phelan, K. McCarthy, and B. Smyth. Clarity at the trec 2011 microblog track. In *Proceedings of the 20th TREC Conference*, Text Retrieval Evaluation Conference (TREC), Gaithersburg, MD, USA., November 2011.

[3] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. *SIGMOD Record*, 29(2):1–12, May 2000.

[4] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on research and development in Information Retrieval*, SIGIR '01, pages 111–119, New York, NY, USA, 2001. ACM.

[5] H. Li, Y. Wang, D. Zhang, M. Zhang, and E. Y. Chang. Pfp: parallel fp-growth for query recommendation. In *Proceedings of the 2008 ACM conference on Recommender Systems*, RecSys '08, pages 107–114, New York, NY, USA, 2008. ACM.

[6] D. Pelleg and A. W. Moore. X-means: Extending k-means with efficient estimation of the number of clusters. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 727–734, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[7] I. Soboroff, D. McCullough, J. Lin, C. Macdonald, I. Ounis, and R. McCreadie. Evaluating Real-Time Search over Tweets. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM 2012)*, 2012.

[8] R. Udupa, A. Bhole, and P. Bhattacharyya. "a term is known by the company it keeps": On selecting a good expansion set in pseudo-relevance feedback. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, ICTIR '09, pages 104–115, Berlin, Heidelberg, 2009. Springer-Verlag.