Vol.2012-DBS-155 No.10
2012/11/19

Automatic query expansion and classification for television related tweet collection

Erik Ward[1,†1,a)] Kazushi Ikeda[2,b)] Maike Erdmann[2,c)] Masami Nakazawa[2,d)]
Gen Hattori[2,e)] Chihiro Ono[2,f)]

Abstract: The growing number of twitter users create large amounts of messages that contain valuable information for market research. These messages, called tweets, which are short, contain twitter-specific writing styles and are often idiosyncratic give rise to a vocabulary mismatch with typically chosen keywords for tweet collection. We propose a method that uses a new form of query expansion that generates pairs of search terms and takes into consideration the language usage of twitter to access user data that would otherwise be missed. Supervised classification is used to maintain precision by comparing collected tweets with external sources. Evaluation was carried out by collecting tweets about five different television shows during their time of airing and indicate, on average a 66.5% increase in the number of tweets compared with using the title of the show as the search terms and 68.0% total precision. Classification gives an average increase of 55.2% in number of tweets and 82.0% total precision. The utility of an automatic system for tracking topics that can find additional keywords is demonstrated.

Keywords: Information retrieval, query expansion, machine learning, twitter, market research.

1. Introduction

The adoption of social media has increased dramatically in the last years. Millions of users use social media services every day, such as many of the 806 million users of *Facebook* [1]. Since the creation of material is decentralized and requires no permission, enormous quantities of unstructured, uncategorized information are created by users every minute. For instance, 340 million twitter messages, often called tweets, are authored every day [2].

Many industries are interested in analyzing this vast amount of user messages where the technologies used include social network analysis and sentiment analysis. One application is to analyze messages about a specific brand, product or similar. We will refer to all such messages as being about a certain *topic*, which has a title, for example, a television shows title.

However, a crucial part of the process of conducting market research on a topic, such as determining sentiment towards it or estimating ratings, is to get a good sample of messages. When gathering messages in social media, often keywords determined by an analyst is used, such as in [3] [4] and [5]. We argue that this method ignores a large fraction of the messages relating to certain topics and thus detrimentally affects the validity of results of later analysis. The idiosyncratic and novel language use on twitter, driven by the short message length, results in a vocabulary mismatch that can be mitigated by the use of a systematic method to find the messages not covered by using the title, or other manually selected terms, as a search terms.

In this paper, to improve tweet collection, we propose the use of streaming retrieval with additional keywords and classification of collected tweets. The additional keywords are determined using relevance feedback techniques and automatic query expansion (AQE). By comparing term distributions in sets of messages about different topics we determine descriptive terms for each topic that yield improved recall when included as search terms. By also classifying the retrieved tweets as either relevant or irrelevant to the topic, higher precision can be achieved. Supervised classification also, in part, deals with the issue of ambiguity [6].

We evaluate the proposed method with regards to tweets about television shows using streaming retrieval for popularity estimation, but the method is not limited to this domain. We used five television shows and collected, on average, 77240 tweets for each show. For each show 500 tweets was sampled randomly, assigned labels and then performance was evaluated.

The rest of this paper is organized as follows: section 2 describes twitter and the options and limitations of retrieving tweets; section 3 lists some related work; section 4 details the methods used; section 5 describes the data used, experiment set up and results; section 6 contains analysis of the results. Finally, section 7 contains conclusions and ideas for future work.

[1] Uppsala University, Lägerhyddsvägen 2, Uppsala, 752 37, Sweden
[2] KDDI R&D Laboratories, Inc. 2-1-15, Ohara, Fujimino, Saitama, 356-8502
[†1] Presently with KDDI R&D Laboratories, Inc.
[a)] erik.ward.5497@student.uu.se
[b)] kz-ikeda@kddilabs.jp
[c)] ma-erdmann@kddilabs.jp
[d)] ms-nakazawa@kddilabs.jp
[e)] gen@kddilabs.jp
[f)] ono@kddilabs.jp

Table 1 Tweets with informal language use.

Alyssa Avila @alyssarenae23
Barney Stinson can sometimes be the romantic type!?????? #HIMYM

Fleur Ozanne @FleuriePoo
Could watch #howimetyourmother for hours

Catarina Heynes @CatarinaHeynes
new episode of #HIMYM in threeeee daysssss!!

Klaroline ? @MeliCont
TVD Production Thanks For TVD #TrendItNow #TVDFamily #TVD

.HOtTeQuiLLa. @Ronnie1596
e19s2 #TVD Katherine's dance :D

Amy Wall @aamy_Wall
#BigBangTheory #NeverFails :)

2. Twitter and tweet collection

Twitter[*1] is a social media service that allows users to share short text messages called tweets limited to 140 UTF-8 characters in length. To a user the messages are presented in inverse chronological order akin to the practice of *blogging* and the short tweets are sometimes called *microblog* posts or status updates.

Perhaps because of their short length users have adopted novel language patterns when writing tweets. One very common practice is the use of *hashtags*, that is, prefixing a word with the # symbol. These often serve as topic markers and some authors [7][8] have defined the inclusion of a certain hashtag as the definition of being related to a specific topic. Also very common is the use of a kind of messaging standard, prefixing an account name with @ refers to a certain user, called a *mention*. Some examples of idiosyncratic language usage patterns on twitter are show in table 1.

The Twitter company allows third parties to access tweets using different methods, one uses persistent HTTP requests in what is called the *streaming API*[*2]. Twitter does not store tweets for long periods of time nor do they support complex search operations such as matching words within a certain proximity or query expansion, instead a Boolean matching strategy is used. A reliable way to access tweets about a certain topic, if we know good search terms, is to sign up to receive tweets containing a disjunction of conjunctions of terms using the *track* function of the streaming API, as opposed to the *fire hose* function that gives a sample of all tweets. In set notation, where t represents a term in the vocabulary Ω, *tweet* is a retrieved message and C_i denotes a conjunction of several terms:

$$API\_results = \{tweet \mid \vee_i \, (C_i \subseteq tweet)\}$$

$$C_i \in \cup_{i,k} \, t_{i,k}, \; t_{i,k} \in \Omega$$

An example:

$$tweet_1 = \{\text{I,like,productX}\}, tweet_2 = \{\text{\#pX,is,bad}\}$$

$$C_1 = \{\text{productX}\}, C_2 = \{\text{\#pX, bad}\}$$

The *track* service is limited to a maximum data rate as well as by $1 \geq \sum |C_i| \geq N$. The data rate is determined by contract and the number of search terms to track, N, is also limited.

[*1] www.twitter.com
[*2] https://dev.twitter.com/docs/streaming-apis

3. Related work

Many authors have investigated information retrieval of tweets, these are mostly adapted to ad-hoc retrieval [9] [10], especially using the TREC microblog data set[*3] [11]. Some authors have employed query expansion such as [12]. In relation to market research, it is an open question weather results achieved on a small data set sampled for a shorter period of time and annotated with a modest number of *query–relevance judgment* pairs are applicable to the problem of obtaining as many as possible related tweets. We are most interested in evaluations done with the constraints of up to date, inclusive tweet collection in place. Nevertheless, many of the techniques used are certainly interesting.

In [13], Mitchell et al. evaluate a system they have set up for on-line television in which social media is integrated. Twitter is used to present tweets about the currently viewed program. Here the twitter API is used and a simple search of the programs title is employed to retrieve relevant messages. Their work represents the basic use of twitter for retrieving TV related tweets and unfortunately recall and precision is not evaluated.

Classification of tweets have been investigated by several authors. Some work with the problem of TV related tweets [4][5] others with other ambiguous topics [6]. However here the test set is collected using simple rules, such as using the title of the topic, or manually selected keywords. A limited form of query expansion is used in [14] to generate the data set, all hashtags found in the data set retrieved by searching for "#worldcup" are recursively used to search for new tweets. In [7] the streaming API is used and messages are classified in a streaming fashion, however the search terms used are manually selected.

Arguing that conventional TV ratings, the so called Nielsen ratings, are outdated Wakamiya et al. employ an alternative method for estimating the number of viewers by counting certain tweets [4]. A large data set collected from the Twitter API during one month was used, where all geotagged [*4] data with Japanese origin available was filtered for the, manually selected, Japanese keywords equivalent to words such as *TV* and *watching*. The key problem of identifying which messages are related to a particular TV shows is addressed and, as seen in other works [6], additional information about the television programs are used: here in the form of an electronic program guide (EPG). Textual similarity is computed between the set of collected tweets and EPG entries. In addition to the textual similarity metric, both temporal and spatial proximity to the television broadcast is used to form a score for each tweet that is compared against a threshold. Experimental results indicate high precision for the proposed method but possibly low recall. Regrettably, no discussion about the statistical significance of the ratings acquired was present.

In a series of papers: [5] [15] [16], a group of researchers from AT&T labs and Leigh University, including Bernard Renger, Junlan Feng and Ovidiu Dan, present a method for classifying ambiguous tweets and an application of their method, Voice enabled social TV. Among other features, the cosine distance from exter-

[*3] https://sites.google.com/site/microblogtrack/
[*4] Some users enable *geotagging* so that the coordinates of the user at the time of posting is publicly available

nal sources are used. Their approach achieves an F-measure of 89%. Their results are only valid as a measure of an overall system if all the relevant tweets can be found using the title of the show as a search term.

4. Proposed method

To increase recall of tweet collection we employ automatic query expansion based on statistics calculated from a large number of tweets. But, increasing recall is not enough for a practical application, one must also ensure that retrieval precision is sufficient. Towards this end we investigate the use of a supervised classifier with the goal of classifying additional retrieved tweets as either being about a TV show, or not.

4.1 Query Expansion

Query expansion is a well known technique in information retrieval (IR) [17] but is often used in IR systems where terms in queries are weighted according to an importance metric. Because we are interested in retrieving data from twitter directly in a streaming fashion we are limited to Boolean search. Therefore we use a slightly modified version of term divergence to gather not terms, but conjunctions of terms.

Following the work of Amati [18] we will use different binomial distributions as our probability space where term frequencies in related documents are considered samples of these distributions. Assume that for each document in a collection it is know whether or not is related to a certain topic. We can then measure the information content of the observed term frequency in the related documents by using a Binomial distribution based on the collection as a whole. For efficiency reasons this requires approximating the binomial distribution using Stirling's formula [19].

$$Inf(t) = F_{t,R} \cdot \mathcal{D}(p_R, p_C) + \frac{1}{2} log_2(2\pi \cdot F_{t,R} \cdot (1 - p_R)) \quad (1)$$

$$\mathcal{D}(p_R, p_C) = p_R \cdot log_2\left(\frac{p_R}{p_C}\right) + p_R \cdot log_2\left(\frac{1 - p_R}{1 - p_C}\right) \quad (2)$$

Where $Inf(t)$ represents the information content of term t in the relevant set. $F_{t,R}$ is the frequency of t in the relevant set, p_R the estimated probability of term t in the relevant set and p_C in the collection as a whole. The divergence function $\mathcal{D}$ is very similar to the symmetric *Kullback-Leibler* divergence.

4.1.1 Co-occurrence heuristic

Instead of producing single term expansion terms we find conjunctions of two terms, u, v as follows: given a list of k terms, check the pairwise co-occurrence of these terms in virtual documents from the relevant set consisting of 5 tweets, the two tweets collected just before and the two collected just after the tweet containing the first term in the conjunction pair. Rank the pairs according to their modified dice coefficient:

$$\tilde{D} = \frac{2 \cdot \tilde{df}_{u \wedge v}}{\tilde{df}_u + \tilde{df}_v} \quad (3)$$

Where $\tilde{df}$ represents document frequency of the virtual documents in the pseudo relevant set and df the document frequency in the collection as a whole.

1: PRS is an array of relevant tweets, $tw_l, 1 \geq l \geq N$.
2: **for all** terms $t \in \cup_i tw_i$ **do**
3: **if** $p_R > p_C$ **then**
4: Use equation 1 calculate $Inf(t)$ and add $\langle t, Inf(t)\rangle$ to list l
5: **end if**
6: **end for**
7: Sort l in order of $Inf(t)$.
8: Let $top[K]$ be an array of terms t_i.
9: $top \leftarrow$ the K terms corresponding to largest $Inf(t)$.
10: return top

Fig. 1 Algorithm, Top(K,PRS), produces an array of single search terms.

1: Let PRS be an array of relevant tweets, $tw_l, 1 \geq l \geq N$.
2: $top \leftarrow$ Top(K, PRS)
3: Let $pairs[K \cdot (K-1)/2]$ be an array of $\langle String, String, Integer\rangle$.
4: **for all** terms t_i in top **do**
5: $T_u \leftarrow$ {tweets $tw \mid t_i \in tw$}
6: **for all** terms $t_j \in top \mid j > i$ **do**
7: $T_v \leftarrow$ {tweets $tw \mid t_j \in tw$}
8: **for all** $tw_l \in T_v$ **do**
9: $vd \leftarrow tw_{l-2}@tw_{l-1}@...@tw_{l+2}$
10: **if** $t_i \in vd$ **then**
11: $\langle t_i, t_j, count\rangle \leftarrow pairs[index(i, j)]$
12: $pairs[index(i, j)] \leftarrow \langle t_i, t_j, count + 1\rangle$
13: **end if**
14: **end for**
15: **end for**
16: **end for**

Fig. 2 Algorithm, Pairs, produces the pairs of search terms used.

4.1.2 Hashtag heuristic

Given a list of k terms, all terms that are mentions or hashtags, start with # or @ respectively, are considered related if the hashtag without the initial pound symbol is not found in a standard English dictionary.

4.1.3 Producing the search terms used

The algorithms in Fig. 1, called Top(K,PRS), and in Fig. 2, called Pairs, show how search terms are generated from collected tweet data using AQE and the co-occurrence heuristic. Top(K,PRS) finds the K most informative terms according to equation 1. Algorithm Pairs finds pairs of terms and their counts of occurrence in the virtual documents. Note that the nested loops on lines 6-14 correspond to doing a join between the tweets that contain t_i and the virtual documents, formed by t_j, that contain t_i. This can be implemented as a hash-join of search results. The final step of sorting the term pairs according to their modified dice coefficient using equation 3 is omitted. The function $index(i, j)$ returns the correct index to store the term pair at in the array $pairs$.

4.2 Classification for improved precision

Since we are interested in increasing recall as much as possible we are not interested in ranking the results of tweet retrieval. Furthermore, because we are working with a stream of results this is not feasible. Using pairs of search terms removes many spurious matches, such as decreasing the probability of a match with a single word from a quote. However, it is to optimistic to assume that all new tweets retrieved by searching for the term pairs created are related and therefore some filtering is necessary.

Table 2 Text sources used for comparing with tweets.

Text source	Description
EPG	Description of show
TV.com	Description of show, character names
Wikipedia page	Main wikipedia page, use of boilerplate algorithm
Top10 Google	The top 10 pages of Google search, use of boilerplate then concatenated
Collected tweets	Concatenation of originally collected tweets containing the title of the TV show
TV words	Television related terms

To increase precision we therefore take our inspiration from related works in tweet classification [15][6] and compare external sources with tweets. The supervised classifier, f, can be seen as a function of two input arguments, a tweet and a show title. If we use K different external sources:

$$c : \mathcal{R}^K \rightarrow \{true, false\}$$

$$f(tweet, title) = c(g(pp(tweet), ws(title)))$$

Here c denotes a supervised, binary, vector based classifier, pp the pre-processing operations listed in section 4.2.1, ws web scraping of external resources as described in section 4.2.2 and g the cosine distance of $tf * idf$ vectors. The features used in c, corresponding to different external sources processed by ws, are listed in table 2. Each source corresponds to one feature in the feature vector that represents the tweet during classification. The feature value is calculated as the cosine distance between the $tf \cdot idf$ vectors of the tweet and the text source. The *TV words* source is not gathered from the web but instead created manually and consists of the words episode, premiere, season, watch, watching and patterns of the form eX, e0X, sX, s0X, sXeX and s0Xe0X with X = 1..10. More accurate document frequencies are estimated using government documents from the American national corpus [20].

4.2.1 Using all information in tweets

In twitter we see several novel uses of the English language, most likely driven by the limit of 140 characters. The following phenomena are present in tweets:

Retweet The letters "RT" before a message indicate that it is a copy of another message.

User tag A unique string associated with each twitter account

Reply and mentions A string of type @[uid] indicates that the message is directed towards a specific user with user tag [uid] or refers to that user.

Hashtags A '#' sign followed by a keyword cam denote the users selected category of the message but we have found that hashtags are commonly used for emphasis as well such as "#bestshowever".

URLs External information is often referenced in tweet using URLs.

To reduce the vocabulary mismatch between tweets and external sources we have employed several pre-processing techniques.

(1) Exchange a mention with the name and description of the user as found using the twitter API.

(2) Split hashtags to the words found in a dictionary with frequency counts where the solution sentence with the highest multiplied frequency of all the words is chosen.

(3) Hashtags that consists of the initial letters of the TV show name are replaced with the show name.

(4) Look up the content of URLs linked from tweets using the boilerplate algorithm and replace the URL with this content, see section 4.2.2.

4.2.2 Web scraping

A lot of content on web pages are not relevant to the main focus of the web page. This content could for instance be commercials or a side menu that offers navigation of the web site and so on. If this non-relevant text was included either as an external source or as additional tweet text found by looking up URLs found in tweets it is likely that the proposed method would be much less effective. Therefore we have chosen to use the *Boilerplate* supervised learning method that has high accuracy when determining informative text sections of web sites [21].

4.3 Complete system

The operation of the system is described in Fig. 3. For processing tweets according to the proposed method several steps are necessary.

- A large corpus of tweets is essential. This means that we need to have ongoing tweet collection for tweets that include titles of TV programs over a longer period of time. During this process tweets are periodically collected and stored in a database. Periodically, the term statistics for each tweet and the corpus as a whole are updated using only tweets that match the filtering criteria of section 5. These statistics are stored in the database called *Term statistics* in the figure.
- For classification external web pages are collected, fed through the boilerplate algorithm, and stored in the database called *External sources*.
- When we have a large corpus of tweets we can train our classifier. By going through the pre-processing steps listed in section 4.2.1, we get a larger body of text representing each tweet. This text is then compared with external sources that were scraped earlier to get the feature vector of each tweet. The process is similar to the process shown in the *Classification* module in the figure but the feature vectors are not used for classification but for training.

With the above preparations done the system is ready to collect new tweets using AQE to improve recall and classification to maintain precision. As can be seen in Fig. 3 the terms generated by the *Query expansion* module are used as arguments for the Twitter streaming API and the resulting tweets are processed by the *Classification* module using a trained classifier model, finally generating the end results. The *Query expansion* module corresponds to the steps listed in section 4.1 where Top(K) and Pairs() refers to the algorithms listed in Fig. 1 and Fig. 2 respectively. The *Classification* module corresponds to the steps listed in section 4.2. The *Pre-process* step corresponds to section 4.2.1.

5. Evaluation

As described in section 4.3 we collect statistics about term distributions in tweets containing the title of TV-shows, train a classifier then perform AQE and classification to evaluate our proposed method.

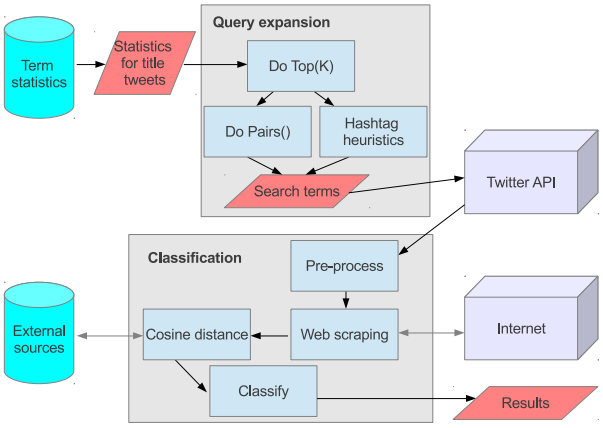

Fig. 3 Collection and classification of new tweets.

Table 3 TV shows used for collecting tweets with new search terms. Shows marked with "*" are aired as reruns multiple times every day.

TV show	Genre	Air times (UTC)
How I met your mother	Drama, Comedy	9/22/*
The big bang theory	Drama, Comedy	9/23/*
The vampire diaries	Drama, Science fiction	9/21/00:00
The X factor	Talent show	9/27/00:00
Wheel of fortune	Game show	9/18/23:30

To get accurate statistics about which terms provide the most information gain using equation 1 we collected tweets containing the titles of 1478 different American TV shows and the most common hashtags found in these tweets, always grouping by the title. Collection has been carried out in excess of six months resulting in more than 133 million tweets. Later we employ strict filtering before calculating statistics to only get tweets which are original and likely to be uniquely about one TV show.

(1) Only keep tweets that contain the title words or a concatenated string of the title words prefixed with #.
(2) Keep only alphanumeric characters and #,@. Remove URLs from consideration.
(3) Remove all tweets containing any capitalization of *RT* as a stand-alone term.
(4) Remove all tweets matching the exact same content as another previously seen tweet.
(5) Remove all tweets that contain more than one show title. This second title must be longer than one word and comes from a list of known shows.
(6) Remove all tweets that are determined not to be English by a naive Bayes classifier.

To evaluate the proposed method we collected data for 5 TV shows of different genres using AQE. Due to limitations of a free twitter API account we could only search for one of these shows at a time and did so for 23h30min staring 6h before airing of the show, see table 3.

To obtain search terms for the twitter streaming API Top(κ) was used with κ=25. Then the hashtag heuristic was applied to get hashtags and mentions as search terms. Finally, the 40 highest ranked term pairs according to equation 3 out of the possible 300 generated by Pairs() was used. For comparison, we also search for the actual title so that we later can filter out all tweets that contain the title to see the increase in number of tweets.

Table 4 Number of tweets collected for the different TV shows during 23h30min.

TV show	Containing title	Extra tweets
How I met your mother	6,271	11,002
The big bang theory	10,222	3,907
The vampire diaries	13,118	23,598
The X factor	62,539	253,376
Wheel of fortune	1,253	912

The system itself is built around a modified version of *Terrier 3.5*[*5] [22] where the language detection used is the open source project *language-detection*[*6].

After obtaining AQE collection results for the different shows a sample of 500 tweets for each show that do not contain the title was labeled. This allows us to see how well the system works without the classification step, see table 8; to evaluate a classifier for the problem, see table 7, and complete system performance in terms of increased number of tweets and precision, see table 9.

5.1 Label criteria

Judging the topic of a message is something most humans are very good at, however this problem is far from trivial. The decision is based upon experience and knowledge of the interpreter about the subject matter itself and the jargon used to talk about it. Consider the following two hypothetical messages:

"When actorX and actorY kiss I get tears in my eyes every time"

"omg #MN is so good, @actorX is the best" where #MN is a hypothetical hashtag used to denote *Movie Name*.

For a person that has seen the movie in question it is obvious that the first message refers to a specific movie. If that person is also an avid twitter user she will understand the second message to be strongly related to the same movie. Much of twitter consists of even more idiosyncratic messages but with the proper knowledge these can be understood and classified.

A strong definition of *related to* is not possible, however we can at least conclude that a message that contains a title that is unique (or almost unequivocally used for one topic) is related. If this title has alternatives in the form of hashtags, messages containing these are also related. Furthermore we can collect messages containing other strongly related meta data terms and leave it up to an evaluator to determine if they are related.

Tweets that are not written in English are manually replaced from the tweets until we have 500 English tweets for each show that are labeled.

5.2 Results

The proposed method gives us a number additional tweets, the results of the experiments when using AQE only are listed in table 4. We can observe that the TV show *The X factor* has an abnormal number of additional tweets compared to the number of tweets containing the title.

Figures 4-5 shows a breakdown of how many tweets per keyword, or keyword pair, were found for the shows *How I met your mother* and *The X factor* respectively. A keyword must account

[*5] www.terrier.org
[*6] http://code.google.com/p/language-detection/

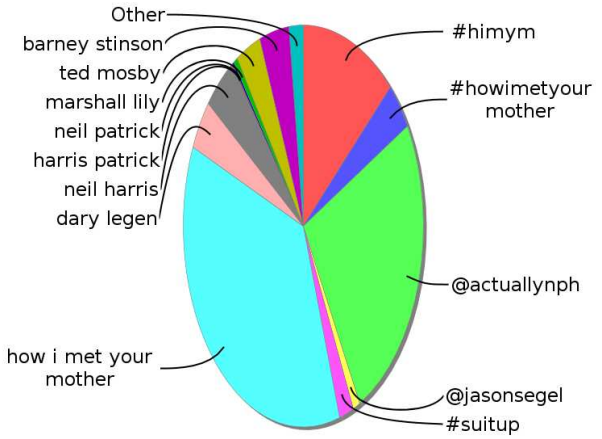

Fig. 4 Fraction of tweets by search terms for *How I met your mother*.

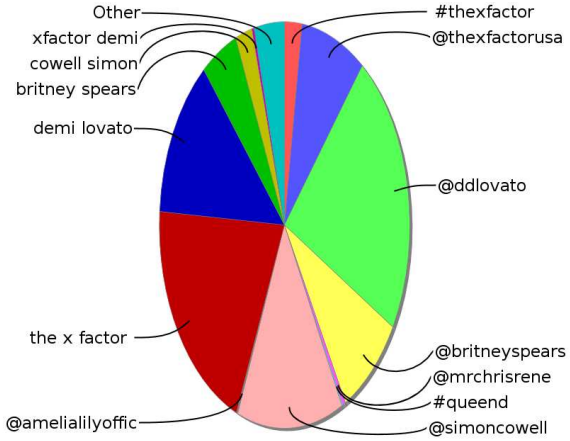

Fig. 5 Fraction of tweets by search terms for *The X factor*.

for at least 0.1% to be included in the chart. These charts show what kind of keywords are generated and how large a fraction of the retrieved results they account for. Most of the keyword pairs do not give many new tweets but a few do. The most important new keywords are arguably different hashtags and mentions. Here we see the reason why *The X factor* has a disproportionate number of additional tweets: the popularity of the celebrity hosts overtake that of the show itself.

To increase precision we wish to remove as many of the unrelated additional tweets as possible. We also want to keep as many as possible of the related ones to achieve our goal of increasing recall. We do this by supervised classification and the chosen algorithm was the J48 implementation of the C4.5 decision tree algorithm using the machine learning toolkit Weka [23].

Best case classification results are listed in table 6 where one model is built for each show and the manually labeled data is used with 10-fold cross validation. The following abbreviations are used: Acc. denotes the accuracy, P_1 the precision, R_1 the recall and F_1 the F-measure. P_1,R_1 and F_1 are calculated for the related class. These metrics are defined as follows, where tp denotes true positive, tn true negative, fp false positive and fn false negative:

$$
\begin{aligned}
Acc. &= (tp+tn)/(tp+tn+fp+fn) \\
P_1 &= tp/(tp+fp) \\
R_1 &= tp/(tp+fn) \\
F_1 &= 2 \cdot P_1 R_1/(P_1+R_1)
\end{aligned}
$$

A feasible system however, cannot rely on manually labeled data and table 7 shows the results when we build one model using assumed labels. The training data is made up of up to 10,000 tweets containing the title for each show that where randomly sampled from a database of collected tweets. These tweets are used both as related and unrelated training examples depending on which of the 5 sets of external sources they where compared against. The test set is composed of the annotated data.

Table 8 shows the class distribution of the labeled sample of 500 tweets that do not contain the title for each show. The table also shows classification results of this sample, indicated by the subscript $_C$. Our classifier is compared to a baseline classifier that assumes all tweets are relevant. In a live system one uses all tweets that are determined to be relevant by the classifier and these correspond to two categories tp_C and fp_C.

After classification we can estimate the performance of the complete system if we assume that the rate related to unrelated tweets is the same for all new tweets that are collected and that the classifier performance is also the same. The maximum likelihood estimation of precision and the increase in number of tweets is calculated with:

$$
\begin{aligned}
\hat{TP} &= |t_{title}| + TP\_rate \cdot P\_rate \cdot |t_{extra}| \\
\hat{FP} &= FP\_rate \cdot N\_rate \cdot |t_{extra}| \\
\Delta tweets &= (\hat{TP} \,/\, |t_{title}|) - 1 \\
prec &= \hat{TP} \,/\, (\hat{TP} \cdot \hat{FP})
\end{aligned}
$$

Here t_{title} denotes the set of tweets containing the title and t_{extra} the set of additional tweets that are retrieved using AQE. The rates P_rate and N_rate is the estimated rate of positive and negative tweets of t_{extra} according to assigned labels. From classification of the labeled data we estimate the classifier performance for all the retrieved tweets with the true positive rate TP_rate and the false positive rate FP_rate. The results can be seen in table 9, where $\Delta tweets_C$ and $prec_C$ are the collection results after classification. Note that for the show *Wheel of fortune* the increase in tweets is actually greater and the total precision lower since not all the tweets containing the title are related, see table 5.

5.2.1 Ambiguity

The issue of ambiguous titles is investigated in [6], [16] and other works. Here we have focused on titles that consists of at least three words and assumed that any tweet that contains all these words is actually about the TV show. To test this assumption we sampled 100 tweets from each show and assigned labels. We can see in table 5 that this assumption is not completely accurate but good enough for our assumption except in the case of *The wheel of fortune*.

6. Analysis

From the results in table 9 we can observe that for one of the shows, *The big bang theory* the precision is high enough to use the tweets without further processing for analysis. For all other

Table 5 Percentage of tweets containing the title that are related to the television show.

TV show	Fraction related
How I met your mother	100%
The big bang theory	99%
The vampire diaries	100%
The X factor	100%
Wheel of fortune	81%

Table 6 Classification results when using manually labeled test data as training data with 10-fold cross validation.

TV show	Acc.	P_1	R_1	F_1
How I met your mother	0.892	0.846	0.856	0.851
The big bang theory	0.894	0.924	0.916	0.92
The vampire diaries	0.784	0.726	0.898	0.803
The X factor	0.876	0.822	0.731	0.774
Wheel of fortune	0.938	0.929	0.954	0.941
Average	0.877	0.850	0.871	0.858

Table 7 Classification results when using training data generated from the same external sources, training examples are from all five shows.

TV show	Acc.	P_1	R_1	F_1
How I met your mother	0.874	0.820	0.833	0.826
The big bang theory	0.886	0.918	0.910	0.914
The vampire diaries	0.746	0.748	0.727	0.737
The X factor	0.508	0.356	0.862	0.504
Wheel of fortune	0.834	0.797	0.916	0.852
Average	0.770	0.728	0.850	0.767

shows except *The X factor* the system precision is adequate. The gains in recall are not dramatic but these are tweets from users that use twitter specific language to express themselves and we believe that it is important to not remove this group since it creates an unnecessary bias. This demonstrates the utility of our system.

For *The X factor* the gains in recall are greater but the precision is not enough and results inspection reveals that tweets about celebrities dominate miss-classifications. It is very hard for the algorithm to separate the actor or television personality from their television appearance, it is also quite hard for a person to do this when assigning labels.

The most effective operational characteristic of the system is the understanding of twitter language use with the help of heuristic methods. Splitting hashtags into their constituent words, looking up web content, resolving user tags used as a substitute for the title and assuming that some abbreviations stand for the shows name allows classification to be accurate. The tweets where this is applicable also correspond to the majority of related additional tweets. A second, much smaller, group of related tweets are not easy to classify correctly, they often refer to events in the shows or voice opinions about how characters or TV personalities behave in the TV program.

This leads us to believe that greater emphasis on which search terms to use is more important than classification. As an example: if the system can understand that the mention @TheXfactorUSA is not about a person but the twitter account associated with the show whilst @ddlovato refers (mostly) to the host of this show as the users idol. Both these users include the string “The X Factor” in their description on twitter.

Matching word two-grams or three-grams from tweets against a full transcript of the show would most likely capture the hard to classify tweets about events that happened in the TV-show, but this requires access to more accurate external data as-well as radically more memory and computing resources. The bag of words assumption made where we compare $tf*idf$ scores is not enough to handle these types of tweets.

Investigating the effects of other features for classification such as correlation with broadcast times is certainly interesting for television related tweets. There is clearly some correlation between the quantity of tweets retrieved with the search terms used and the broadcast time, see Fig. 6.

7. Conclusions and future work

We performed AQE using a large corpus of specially collected tweets containing television titles to produce new search terms. These search terms where then used to gather data directly from twitter and an increase in number of tweets was estimated for five different television programs. The average increase in number of tweets was estimated to 66.5%.

To improve precision a classifier was used on the gathered data that did not include the original title, this classifier uses web-scraping to compare tweets with external sources with mixed results, for three of the five television shows the gains in recall are modest but the precision is high enough to consider the system functional. For two shows the increase in number of tweets retrieved is good but the false positive rate is to high for accurate analysis, this is especially true for the TV show *The X factor*.

In short, there is evidence that an automatic system can find additional keywords for more effective market research using tweets.

Future work will include investigating the effects of the different parameters of the method such as the number of tweets used to form *virtual documents*, the number of terms used and the number of term pairs generated. Furthermore, it should be possible to rework the hashtag and mention heuristic to improve results. Regrettably, the lack of an annotated product tracking corpus of tweets makes it very time consuming to evaluate these parameters and perhaps it is best to focus on theoretical analysis.

The bag of words assumption used when comparing to external sources does not capture all of the most difficult tweets to classify, those that do not contain some version of the title. This warrants investigation into other methods to increase precision of the method.

Investigating the use of the system for ratings calculations and correlating with conventional methods is certainly an interesting avenue for future work.

Acknowledgments Erik Ward wishes to thank the Göran Holmquist Foundation and the Sweden Japan Foundation for travel funding.

References

[1] B. Block.
“Facebook: Around the world in 800 days,” May 2012.
http://blog.comscore.com/
Accessed 10/10/2012

[2] Twitter turns six, Mar. 2012.
http://blog.twitter.com/2012/03/twitter-turns-six.html
Accessed 10/10/2012.

[3] A. Tumasjan, T. Sprenger, P. Sandner, and I. Welpe, “Predicting elections with twitter: What 140 characters reveal about political senti-

Table 8 Class distribution of annotated data after classification by baseline and C4.5 classifiers.

TV show	tp	fp	acc. (new)	tp_C	fp_C	$acc._C$ (new)
How I met your mother	180	320	36.0%	150	33	82.0%
The big bang theory	334	166	66.8%	304	27	91.8%
The vampire diaries	245	255	49.0%	178	60	74.8%
The X factor	145	355	29.0%	125	226	48.3%
Wheel of fortune	261	239	52.2%	239	61	79.7%

Table 9 System performance, before and after classification.

TV show	$\Delta tweets$	$prec$	$\Delta tweets_C$	$prec_C$
How I met your mother	63.2%	59.2%	52.6%	88.5%
The big bang theory	25.5%	90.8%	23.2%	97.7%
The vampire diaries	88.1%	67.2%	64.1%	83.2%
The X factor	117.5%	43.1%	101.2%	48.3%
Wheel of fortune	38.0%	79.9%	34.8%	91.4%
Average	66.5%	68.0%	55.2%	82.0%

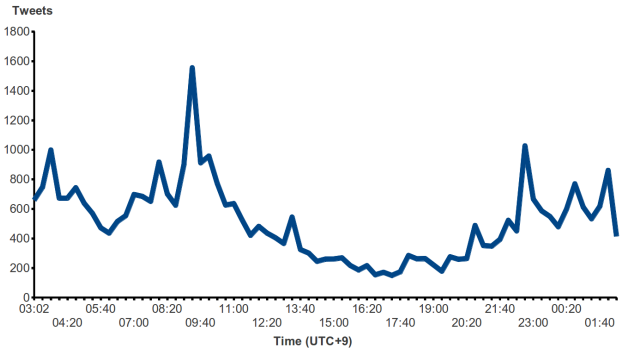

Fig. 6 Histogram of number of tweets by 20 minute periods during the collection of tweets about *The vampire diaries*. Table in UTC+9h starting from Sept. 21 and ending Sept. 22 2012.

ment," 2010.

[4] S. Wakamiya, R. Lee, and K. Sumiya, "Towards better TV viewing rates: exploiting crowd's media life logs over twitter for TV rating," in *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication*, ICUIMC '11, (New York, NY, USA), p. 39:139:10, ACM, 2011.

[5] B. Renger, J. Feng, O. Dan, H. Chang, and L. Barbosa, "VoiSTV: voice-enabled social TV," in *Proceedings of the 20th international conference companion on World wide web*, WWW '11, (New York, NY, USA), p. 253256, ACM, 2011.

[6] S. Yerva, Z. Miklós, and K. Aberer, "It was easy, when apples and blackberries were only fruits," in *Third Web People Search Evaluation Forum (WePS-3), CLEF*, 2010.

[7] K. Nishida, R. Banno, K. Fujimura, and T. Hoshide, "Tweet classification by data compression," in *Proceedings of the 2011 international workshop on DETecting and Exploiting Cultural diversiTy on the social web*, DETECT '11, (New York, NY, USA), p. 2934, ACM, 2011.

[8] C. Wagner and M. Strohmaier, "The wisdom in tweetonomies: acquiring latent conceptual structures from social awareness streams," in *Proceedings of the 3rd International Semantic Search Workshop*, SEMSEARCH '10, (New York, NY, USA), p. 6:16:10, ACM, 2010.

[9] J. Teevan, D. Ramage, and M. R. Morris, "#TwitterSearch: a comparison of microblog search and web search," in *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, (New York, NY, USA), p. 3544, ACM, 2011.

[10] M. Efron, "Information search and retrieval in microblogs," *Journal of the American Society for Information Science and Technology*, vol. 62, no. 6, p. 9961008, 2011.

[11] S. Bhattacharya, C. Harris, Y. Mejova, C. Yang, P. Srinivasan, and T. Track, "The university of iowa at trec 2011: Microblogs, medical records and crowdsourcing,"

[12] K. Massoudi, M. Tsagkias, M. de Rijke, and W. Weerkamp, "Incorporating query expansion and quality indicators in searching microblog posts," in *Advances in Information Retrieval* (P. Clough, C. Foley, C. Gurrin, G. Jones, W. Kraaij, H. Lee, and V. Mudoch, eds.), vol. 6611 of *Lecture Notes in Computer Science*, pp. 362–367, Springer Berlin / Heidelberg, 2011.

[13] K. Mitchell, A. Jones, J. Ishmael, and N. J. Race, "Social TV: toward content navigation using social awareness," in *Proceedings of the 8th international interactive conference on Interactive TV&Video*, EuroITV '10, (New York, NY, USA), p. 283292, ACM, 2010.

[14] K. Ariyasu, H. Fujisawa, and Y. Kanatsugu, "Message analysis algorithms and their application to social tv," p. 1, ACM Press, 2011.

[15] O. Dan, J. Feng, and B. Davison, "Filtering microblogging messages for social tv," in *Proceedings of the 20th international conference companion on World wide web*, WWW '11, (New York, NY, USA), p. 197200, ACM, 2011.

[16] Dan Ovidiu, Junlan Feng, and Brian D. Davidson, "A bootstrapping approach to identifying relevant tweets for social TV," in *ICWSM*, (Barcelona), 2011.

[17] C. Carpineto and G. Romano, "A survey of automatic query expansion in information retrieval," *ACM Comput. Surv.*, vol. 44, p. 1:11:50, Jan. 2012.

[18] G. Amati, "Glasgow university theses repository - probability models for information retrieval based on divergence from randomness." http://theses.gla.ac.uk/1570/, 2003.

[19] A. Renyi, *Foundations of probability*, vol. 9. Holden-Day San Francisco, 1970.

[20] N. Ide and K. Suderman, "Integrating linguistic resources: The american national corpus model," in *Proceedings of the 6th International Conference on Language Resources and Evaluation*, 2006.

[21] C. Kohlschtter, P. Fankhauser, and W. Nejdl, "Boilerplate detection using shallow text features," in *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, (New York, NY, USA), p. 441450, ACM, 2010.

[22] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma, "Terrier: A high performance and scalable information retrieval platform," in *Proceedings of the OSIR Workshop*, pp. 18–25, Citeseer, 2006.

[23] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, p. 1018, Nov. 2009.