Opinions are like voices, we all have a different kind:
Pop concerts and their effect on user generated content

V.J. Oord
0104744
voord@science.uva.nl

Thesis Master Informatiekunde
Programma Human Centered Multimedia

Universiteit van Amsterdam
Faculteit Natuurkunde, Wiskunde en Informatica

28/08/2007
Begeleiders: Maarten de Rijke en David Ahn

Maarten de Rijke,

David Ahn,

OPINIONS ARE LIKE VOICES, WE ALL HAVE A DIFFERENT KIND: POP CONCERTS AND THEIR EFFECT ON USER GENERATED CONTENT

Vincent Oord

Universiteit van Amsterdam
voord@science.uva.nl

ABSTRACT

In this thesis a system which automatically retrieves user generated content related to pop concerts will be discussed. User generated content is a popular item in todays web environment consisting of weblog posts, photos and video clips that internet users publish online through various channels. Because of the abundance of user generated content on the web today, people have great difficulties finding relevant information. This thesis describes both the retrieval proces as the classification proces that underpin this system. An experiment is conducted using different kinds of features and machine learning algorithms to evaluate the most optimal settings for such a system. Using a small set of features and a memory based learner the system obtains over 85 percent accuracy on a set of Dutch weblog post and newspaper articles.

1. INTRODUCTION

Weblogs can be seen as a source of information and commentary about current events [1]. A source that has grown enormously in the last couple of years and which continues as of today. BlogPulse, a search engine targeted at weblogs[1] currently identifies over 55 million different weblogs, with almost 100,000 new weblogs popping up every 24 hours. The mentioning of weblogs in today's media and society is so ubiquitous that the word is in the top list of hated Internet words.[2] Nevertheless, the power of weblogs is being recognized by scholars and marketing people all over the world. Never before has such a large, freely accessible repository where so many people discussed all kinds of topics existed. These opinions can be very valuable to companies and marketeers, as they can be used to predict the succes of certain products or services, like movies [2].

But the participation of normal people on the web is not limited to the use of weblogs. More and more websites rely on their users to create and share content. From passive consumers people are transformed to active participants. This goes beyond expressing feelings in texts and can include tagging data, sharing knowledge and publishing photos and movies. This so called *user generated content* can be exploited by creating systems that aggregate this data and use it as input for machine learning tasks, thus creating some sort of artificial intelligence which uses the web as a large brain. [3]. Usage of these systems includes product recommendations [4], collaborative filtering [5], or automatically determining the author or genre of a piece of music [6].

User generated content is not only valuable to companies and scholars. The success of so called social sites which promote sharing of user generated content lies not only in the fact that people like to create content, but that they also like to view the content of others. A huge problem for these people is that because of the abundance of social websites and services who offer user generated content, it can be difficult to find the things they are looking for. Systems that are targeted at finding specific information can help them in finding what they find, easily.

In this thesis the first steps towards developing such a system are discussed. The main point of the system is to help a user find pieces of user generated content related to a particular concert. People who visited a concert like to read if other people share their opinion of the concert. Due to the great supply of websites users can post their reviews to, it can be hard to find many reviews. It would be nice if a great part of all these websites were automatically scanned and sought for concert reviews and then aggregated in one place, which the user can then visit to read a great number of reviews of other visitors to that particular concert.

The goal of this thesis is to learn some lessons for the efficient retrieval of user generated content related to pop concerts and to find good features for the classification of these reviews. The main research question addressed here therefore is:

> *In what way do cultural expressions like pop concerts have a by-effect on the presence of user generated content and how can this content be captured in an easily accessible web environment?*

I would like to thank the following people for supporting me in this work: Rob van der Zwaan of PodiumInfo.nl, my supervisors Maarten de Rijke, David Ahn and Erik Tjong Kim Sang, who helped me with the last bits of the thesis when David moved back to the U.S.A., and last but not least my friends & family. A digital copy of this thesis can be found at http://science.uva.nl/~voord/docs/ma-thesis.pdf

[1] http://www.blogpulse.com

[2] http://www.chinadaily.com.cn/world/2007-06/22/content_899724.htm

The remainder of this thesis is organized as follows. The next session discusses some background information on the research topic, such as related research, problems common for the domain and some definitions. Section 3 focuses on the retrieval part of the task described, while section 4 is about automatically classifying concert reviews. In section 5 the presentation of the data is explored. The final section , section 6 gives an overview of the findings of this thesis and some pointers for future research.

2. BACKGROUND

This section covers some background information on the topic being researched: difficulties specific for the domain used, definitions of terms used in the thesis and a description of the system architecture and required features.

One of the main difficulties of using weblog texts to find things like concert reviews is the fact that these texts are written by amateurs, with all the drawbacks that come along. For instance, weblogposts are not always neatly structured, can have more than one topic and are not edited which makes them sensitive for typos. All these factors have their influence on the quality of systems that use weblogs as their sources [5].

2.1. Definitions

For the purpose of clarity some of the terms used in this thesis are defined here. First, the term user generated content (UGC), which is sometimes also referred to as community metadata or consumer generated media. UGC includes all forms of media, whether it is text, photography, video or audio that is created and put online by members of the public using weblogs, forums and social websites. These people have usually little or no professional background in creating or publishing media. In the scope of this paper, UGC is also content placed online by professional media, like newspapers or particular music websites run by professional journalists. Ideally, the system described in this thesis would include multimedia UGC, but due to time and space restrictions it is limited to just texts.

Another frequently used term in this thesis is 'band'. By this term, any musical formation is meant, either solo singers, dj's, combinations of those or groups of musicians. Whenever a band or band-name is mentioned, this can be read as an artist, or as an artist name.

2.2. Features

There are three important features any system that retrieves information like user generated content from the web should have in order to be successful, as identified by Evans [7]: speed, robustness and relevancy.

The first feature, speed, is important with respect to how fast data can be processed by the system. In web retrieval corpora of several gigabytes are not uncommon, so data processing should be as efficient as possible. Speediness of data processing becomes even more important when the targeted system is constantly updated with new data, e.g., new weblogposts. In this thesis the system will only be using historical data which is indexed once and therefore speed is less of an issue.

As a second feature, robustness is related to the fact whether the system is able to cope with the dynamic nature of the web in general and blogs in particular, e.g., various ways of spelling a band or artist's name (both 'Red Hot Chili Peppers' 'Peppers' and 'RHCP' refer to the same band) and typing errors ('Amy Whinehouse', 'Amy Winehouse').

Relevancy, is the third important feature the system should have. This feature is about the nature of the to be retrieved documents and is of course context dependent. In the system described in this thesis relevant documents are any user generated content that is review-like or report-like - thus subjective in nature - which are about a pop concert preferably ordered by date so documents can be matched with specific concerts. This means that there are actually two classification tasks, one on topicality, i.e. is the text about a concert by a band, and one on orientation, i.e. is the text subjective in nature. Documents which could show up but are not seen as relevant within the context of this application are announces of concerts, concert agenda's and news items, e.g., about a band splitting up. A drawback in finding relevant documents in a large unstructured corpus like the web is ambiguity, for instance in band names. A lot of bands have very generic names like 'Texas' 'Editors' or 'The Killers' which will likely increase the noise in the document set related to one of those bands. An easy solution to account for this is to add certain keywords like 'music' or 'review' to a search query [5, 8]. In short, the system should be able to make clear distinctions between documents that are relevant to the task at hand and those that are not. This will be explained further in section 4.

In addition to these three features, a fourth important feature not mentioned by Evans [7] should be mentioned. This is the degree of extendibility of the system. Input of the system depends on agendas of clubs and each club might have its own way of publishing its agenda so the system should be able to handle different kinds of formats to extract all the useful information from these agendas. Not all clubs will have a nice machine readable version like an XML file of their agendas online, so some of this information will have to be scraped from their websites. The use of wrappers for this task might be useful, as most agendas do have some sort of similar format, i.e they show the date and time of the concert, the name of the band and the location. Another part of the system where extendibility plays an important role is in the sources used for retrieving user generated content. It would be very useful if the system is able to find new sources on the web which contain music review and automatically be added to the system. Finally, adding other types of user generated content like pho-

tos, movies and audio to the system could also be very useful, so the system should be prepared for this as well.

2.3. Architecture

With the aforementioned features in mind a system architecture has been designed for the proposed system. In this section this architecture will be discussed and the parts on which the focus of this thesis lies will be highlighted. The components of the system and information flow through the system is shown in figure 1.

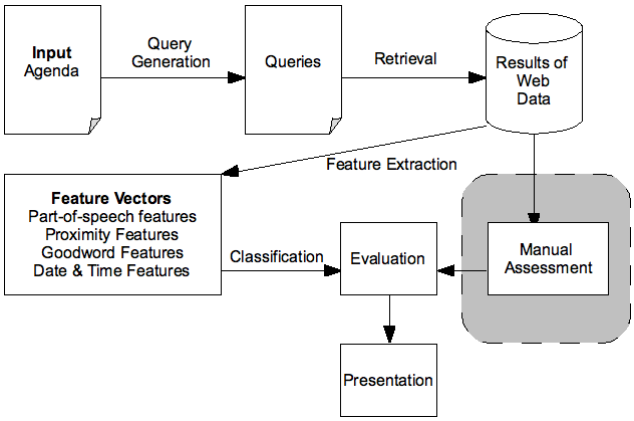

Fig. 1. System Architecture

The first component is the input component, which is an aggregate of concert agendas of different venues. Based upon the information in this component various queries are generated which then can be used to retrieve information from a data source, like the internet or a previously crawled corpus of various text sources. Each text is then analyzed and the features that are extracted are used as input for the classification proces. Any texts that are classified as relevant (i.e. contain a concert review) are then stored with respect to the concert they are about and finally presented to the user in the presentation component. Figure 1 also shows a component called 'Manual Assessment', which is separated from the rest of the architecture, because this component is only used in the initial learning phase of the system. In this phase, results from the queries are assessed by a human in order to provide the classifier a 'ground truth' which is used to train the classifier and evaluate its results. When the system is fully operational, this component is not needed anymore.

3. RETRIEVAL

In this section the process of information retrieval is described, which is the first important step in developing a concert review display system. First, the necessary input in the form of a concertagenda is obtained, then documents relating to the events in this agenda are searched for. These results are then manually assessed to be used as training data for the classifier discussed in section 4.

3.1. Data

For the concertagenda a RSS feed of a Dutch concert agenda aggregation site named Podium Info[3] is used. This RSS feed is a custom made XML document consisting of all pop events from June 2006 that took or will take place in so called A-location clubs in the Netherlands. A-locations are large venues with a capacity of 1000 people and more and include the Heineken Music Hall, Ahoy, Paradiso and large football stadiums like De Kuip and Amsterdam ArenA. Using these premium locations means that most bands will be relatively well known which should result in sufficient user generated content. The use of an XML file makes it possible to easily extract important information for each event, like the date, name of the band, name of the venue and name of the city, without the need of writing wrappers to scrape this kind of information from the websites of clubs. Based upon this information queries are generated which are then used to retrieve related user generated content.

Retrieval is done based on a readily available corpus of newspaper articles from all large Dutch newspapers and weblogposts from Dutch weblog provider web-log.nl gathered during the Verkiezingskijker project [9], (from now on 'VK-set'). It covers all articles and weblogposts on a variety of topics published as from June 2006 until recently. Actual retrieval of relevant documents is done using various query methods which are described in the next section.

3.2. Query Assessment

For this particular retrieval task a query based approach to retrieval is chosen, similar to the approaches in [8] and [5]. A query based approach is a simple and quick method which yields adequate results compared to a crawling approach with would be more time and resource expensive. As mentioned in section 2, there are a lot of difficulties with retrieving pop music reviews in user generated content - as with any other retrieval task - due to generic and thus ambiguous terms, in this case often band names.

To overcome the problems of ambiguous terms in queries a couple of different query types were tried and evaluated in an assessment experiment to see which type of query retrieves the most relevant documents. Two-hundred events were randomly chosen from the Podium Info agenda and for each event two query types were generated, a phrase query and a keyword query. Each of these types of queries were constructed in two different ways, using the band name and the city or the name of the club where the event took place. In phrase queries both band name and city or venue were writ-

[3] http://www.podiuminfo.nl

ten between quotes, in keyword queries each term was seen as a separate keyword (see figure 2 for examples).

Little Man Tate Amsterdam
Little Man Tate Paradiso
”Little Man Tate” ”Paradiso”
”Little Man Tate” ”Amsterdam”

Fig. 2. The four different types of queries for each event

Because each query is executed on both the weblog and newspaper data, this leads to a total of of 1600 queries from which the first ten results were manually assessed. In practice much less results are assessed, since not all queries did return ten or more results. As a matter of fact, only 302 queries out of 1600 return ten or more results, while 777 queries returned zero results. In total 4631 documents are returned to the 1600 queries that are executed: 3483 weblog posts and 1148 news articles. Out of these, 1430 (41.1%) weblog posts and 801 (69.8%) news articles are unique documents.

Table 1 shows some statistics of the results of the different queries. On average, queries executed in blog posts yield more documents then queries executed in news, while using the name of a venue of a concert to enrich a query yields less documents than the name of a city. Phrase queries also return less relevant documents than keyword queries. With respect to the amount of documents being returned the best query to use is a keyword query, enriched with the name of the city where the concert took place (kw_city) and executed in the weblog domain. But this not necessarily means that this type of query retrieves the most relevant results, as appeared after assessing the results.

	news	*blogs*
keyword queries	40.4	60.6
phrase queries	2.1	7.9
city queries	41.8	57.8
venue queries	0.6	10.7

Table 1. Average Hits per Query Type

Results were assessed by using a four point scale from zero to three. Each score corresponds to a label, as can been seen in table 2. A documents is labelled ‘is_event’ only if it is a review about the concert which the query was looking for. When a document is a review about a concert, but not the one from the query, e.g., another location or another date, it was labelled ‘is_concert’. Documents that are about the band, but did not contain a concert review are labelled ‘is_band’. All other documents are labelled ‘is_nothing’. Because evaluations are made with respect to the query, some documents labelled as ‘nothing’ could in fact be concert reviews but one of a different band than the query asked for. Because this occurs only rarely, the risks of any bias in the assessments are limited.

Score	*Label*	*Meaning*
0	is_nothing	unrelated document
1	is_band	about the band, no concert
2	is_concert	other concert
3	is_event	the concert from the query

Table 2. Assessment Labels

Most of the documents found with respect to the executed queries appeared to be not very relevant after all, as can be concluded from the statistics in table 3: a majority of the documents (76.3%) is labelled with ‘is_nothing’. There are various reasons for this. First, newspaper articles often state the name of the city from which the news is being reported in the beginning of the article, so when using city queries a lot of these articles show up. Furthermore, a lot of news articles about cities or about sports teams from those cities appear in the results when using city queries (e.g., ‘FC Utrecht’, a football team). Second, a lot of returned documents returned contained nothing more than concert agendas, or a list of upcoming events, or other varieties of lists of bands and artists. A third reason can be found in the use of generic band names. While this is a Dutch corpus and a lot of band names are in fact English, this effect is much smaller than it would be in English corpora, it did happen in some occasions, for example with the Dutch band ‘De Dijk’, with has the very common word ‘dijk’ (dike) in its band name. A fourth and last reason is similar to the phenomenon observed in [4] and has to do with references to bands in documents about other bands, or weblogposts who mention the song a blogger is currently listening to.

From table 3 it is clear to see that many more concert reviews can be found in weblog posts than in newspaper articles. This can be explained by the fact that newspapers have limited space reserved for cultural events. Moreover, it is very well possible that not all of these reviews that appear in print are also published online. Weblogs do not have these constraints: bloggers usually put their reviews online within days or even hours of a concert. The only exception to this are the phr_venue queries in the news domain, which returns a lot of relevant results. It should be noted however that this particular query returns very few documents in total.

While k_cit queries are a good choice from a number-of-hits perspective, table 3 shows that these queries do not return the most relevant documents. The use of phrase queries, enriched with the name of the venue (p_ven) seem to be a lot more efficient, with almost 45% of the post being relevant in some way (i.e. labelled as ‘is_event’, ‘is_concert’ or ‘is_band’). Overall this means that from a retrieval point of view it is best to only use blog posts as a source and phr_venue type of queries to retrieve documents, as this combination has

the highest precision of all methods. A combination of methods can be used however to improve recall.

Query Type	*Hits*	*3*	*2*	*1*	*0*
Blogs					
phr_city	629	7.9%	2.7%	21.3%	68.0%
phr_venue	477	13.0%	8.2%	22.9%	56.0%
kw_city	1342	3.7%	1.9%	12.6%	81.7%
kw_venue	1035	6.7%	3.7%	12.0%	77.7%
News					
phr_city	173	1.7%	7.5%	23.7%	67.1%
phr_venue	45	15.6%	15.6%	51.1%	17.8%
kw_city	787	0.5%	1.3%	6.9%	91.4%
kw_venue	143	2.1%	9.1%	22.4%	66.4%

Table 3. Assessment Results for Blog and News Items

4. CLASSIFICATION

The next important step in the development of the system is the classification phase, which makes it possible to automatically determine whether a retrieved document contains a review about a particular concert or not. The documents that are classified as being review like can then be stored and presented to the user. More on that in the next section.

In order to train the classifier successfully, some 'ground truth' about what kind of documents are seen as relevant (i.e. review like) and which are not is necessary. This ground truth is available through the manual assessment of the results found in the retrieval phase of development. The disadvantage of these assessments however is the nature of the documents with a 'is_nothing' label. Because each document is assessed with respect to the query it belongs to, some of the documents labelled as 'is_nothing' actually are concert reviews, but just not of the right band. So using documents with this label as negative examples could cause low accuracy for the classifier as there is not always a clear distinction between documents with the 'is_nothing' label and the 'is_event' and 'is_concert' labels.

Therefore, all the documents labelled as 'is_nothing' are not used in training the classifier. Instead, documents labelled as 'is_event' and 'is_concert' are merged together to form the class of positive examples, since these labels consist only of documents that are concert reviews. Documents labelled as 'is_band' are never concert reviews and usually news like articles about a band or artist and they are used to form the negative examples class.

This results in a total dataset of 1097 documents consisting of 411 positive examples and 686 negative examples. The documents have an average length of 575 terms, the largest document having an length of 8095 terms, the smallest document having a length of only 4 terms.

4.1. Baseline

First a simple baseline experiment is conducted using a simple bag-of-words method. Bag-of-word is a very easy and quick way to represent documents in which all terms from a document are seen as possible features according to the intuition that the words used in documents are very descriptive for the class to which the document belongs.

In this case, all terms from the document collection are extracted without removing any stopwords and used as a binary feature, meaning that that a term either is present or not in a document. Terms occurring more than once in a document are thus treated in the same way as terms that occur only once. This is indeed a very naïve approach to a document representation, but good enough for baseline purposes as it is only needed to evaluate the value of using other more complicated and expensive features.

For classifying the results of the baseline a SVM classifier is used [10]. Support Vector Machines (SVMs) are very well suited in natural language processing (NLP) tasks, outperforming other classifiers [11]. With this method feature vectors are mapped to a high dimensional space and a hyperplane is constructed to separate the data. Unlike regular linear classifiers, which often use any possible separation between data points, this hyperplane is aimed to create the most optimal separation between the data points in order to minimize classification errors. Apart from that, SVMs are also very suitable to handle a large amount of features, which is not uncommon in NLP tasks.

Because of the relatively small size of the dataset training the classifier is done using tenfold cross-validation, meaning that the dataset is divided in ten folds of roughly the same size so the classifier can be trained with nine folds and tested with the remaining one fold. This process is then repeated for all the folds and the results for each train and test run are then evaluated.

With an average accuracy of 85.0%, a precision of 89.2% and a recall of 68.1% the baseline performed surprisingly well. This can be due to the fact that a lot of examples in the dataset are negative and the classifier tends to classify unknown instances as a negative example. The low recall supports this finding. Other features should be mainly aimed at improving recall while maintaining similar precision and accuracy.

4.2. Experiment

From the baseline experiment it can be seen that using a simple approach a relative high score can be gained. To improve this score, and especially recall, another feature vector representation of the document set is created and classified using the same SVM classifier. In this section first these features will be discussed in more detail. Results of the experiment follow after that discussion.

4.2.1. Features

Three different types of features are used. The first type is a bag-of-words method, based on statistics of frequently occurring terms. Table 4 shows the top ten terms for positive and negative posts and their occurrences.

Positive		*Negative*	
Term	Frequency	Term	Frequency
band	450	band	1733
concert	350	forum	1722
goed	303	voorpagina	1706
jaar	295	album	1664
erg	268	jaar	1233
optreden	244	eerste	703
zaal	234	bron	645
keer	229	single	576
uur	221	twee	569
leuk	218	bands	533

Table 4. Term Occurrences for positive and negative posts

Term	*Frequency*	*Term*	*Frequency*
producer	130	contract	72
downloaden	106	amerika	71
releasedatum	94	friedman	70
videoclip	90	burnt	70
ruben	90	popov	68
voorronde	87	bevestigd	66
fonds	86	youtube	64
paaspop	82	soundslike	64
york	75	popprijs	64
bad	74	geruchten	64

Table 5. Top 20 Unique Negative Term Occurrences

As is clear from the table, a lot of terms occur very often in both positive and negative texts, reducing the number of terms that are usable as a discriminating factor between positive and negative instances of texts. Therefore two new top lists are constructed, this time only with terms that are unique for positive or negative texts. Out of these two lists, only the negative top list is usable, because the frequencies in the positive top lists are all below thirty, which is too small to be representative for positive posts. In the negative list, the top twenty terms all occur sixty times or more, which makes this list a good source for some simple bag-of-word features. Each term is used as a binary feature, much like in the baseline experiment. Table **??** shows these terms and their frequencies.

This doesn't mean that the top lists from table 4 are totally useless, because term frequencies can always be useful in text classification. Hence, the top lists are used as cumulative features, i.e. for each text the number of terms that appear in the top lists is counted, thus resulting in four additional features (frequently occurring adjectives and nouns in both positive and negative texts).

The second type of feature are date and time features. They consist of two features that are derived from the date and time properties of both events and texts. The first one takes the number of days between the event and the time of publishing of the weblogpost or newspaper article. Usually, as can be concluded from the manual assessment of query results from section 3.2, concert reviews are posted within days of the concert, so positive instances should have a low value for this feature, while negative instances should have a very high, or negative, value.

But posting time is not always a clear pointer to tell if a text is a review or not. During the manual assessment task it became apparent that a lot of posts, relating to concerts are posted at the end of the year. This is especially the case with weblogs, as bloggers looks back at the concerts they in the past year and comment on them. Hence, the 'end of year' feature is also included, to account for this phenomenon.

The third and last type of feature used is a proximity feature in combination with part-of-speech (POS) tagging for Dutch language [12]. The goal of this feature is to exploit information which is in close proximity of a particular named entity or subject within a text, under the assumption that the terms in the close vicinity of a band name give a good indication whether that part of the text is subjective, thus likely to be a review, or objective, thus likely to be something else. The orientation of a textpart is determined through tag the text using a POS-tagger and counting the number of adjectives and personal pronouns that occur in the text.

Extracting textparts out of each weblogpost or newspaper article can be difficult because band names are not always spelled uniformly within one text, let alone across different texts. Weblogs tend to have this problem more often than newspaper articles as their writers rarely are professionals and do not spellcheck their entries or refer to bands using abbreviations. Although it is hard to account for all possible permutations of a band name, these variations have been taken into account while locating a band name in a text. First, the article of the band name is removed (either Dutch articles like 'de' and English articles like 'the') because these articles are sometimes omitted while referring to a band. Then, each term of the band name is looked for and for each hit it is checked if the term occurring first after the found band name term is also present in the band name. The results are summed and if more than one band name term occurs successively in the part of the text, it is seen as a occurrence of the band name in that part. This helps improving recall in a text about, for instance, the band 'Michael Franti & Spearhead', when the author of the text mentions only 'Michael Franti' or when he or she uses 'and' instead of '&'.

When the name of the band is located in the text, a k number of terms around the band name are extracted and analyzed.

Choosing a right number for k can be difficult because the orientation of a text can change when using a bigger or smaller window of terms around the band name [2, 13]. In this experiment k was set to twenty, setting the proximity window to twenty terms before and twenty terms after the band name. In case there were less then twenty terms before or after the band name the number of remaining terms was used. Multiple occurrences of a band name in a text mean multiple proximity windows. Figure 3 shows an example of a proximity window with $k = 15$.

ook nog even de grote trommel mee omhoog houdt kan je avond niet meer stuk **michael franti en spearhead** ook al heeft hij beatboxer radioactive niet bij speelt hij vrijwel geen

Fig. 3. Weblog post showing band-name and k terms around it

Each proximity window is then tagged with a part-of-speech tagger. Part-of-speech tagging is a process in which terms from a text are automatically analyzed and 'tagged' with the part-of-speech they belong to (e.g., nouns, adjectives, verbs). In this experiment, it is used to count the number of adjectives and personal pronouns that are present in the proximity window. Underlying assumption is that in concert reviews these types of words are used more frequently than in non-reviews, like news articles or concert announcements. Since any type of review is sought for in this application, semantic orientation, i.e. the fact whether a text is positive or negative, is not relevant in this case. Only determining subjectivity is important, hence only the number of adjectives and personal pronouns is used. To account for band name occurrences at the beginning or the end of a text who do not have k terms in its vicinity, these scores are normalised by dividing the number of relevant part-of-speech terms by the total number of terms in the window. When multiple windows are present in a text, the average for all the windows in the text is taken.

4.2.2. Results

Classification was done using SVMs and tenfold cross-validation, similar to the settings in the baseline experiment. Using the described set of features 702 of 1097 instances were correctly classified, meaning an average accuracy of 63.9%. Precision was at 76.7% with a recall of only 5.6%. In this context, precision is the amount of true positive instances among those that are labelled positive while recall is the number of positive instances that are classified as positive in relation to the total number of positive instances in the entire collection. Just as in the baseline experiment, the classifier tends to classify any instances as negative examples, but in this case the effect is even stronger.

From these results it could be concluded that the chosen feature-set is not as good as expected as it results are significantly lower than the results from the baseline experiment. Another option is that given the before mentioned feature-set SVMs is not the best suited classifier for this task. To test these hypotheses a new classification experiment is conducted, this time using another classifier, TiMBL [14].

4.2.3. Changing Classifiers

TiMBL (Tilburg Memory Based Learner) is a machine learner which combines different types of memory based classification algorithms [14]. Memory based learning (MBL) is related to k-Nearest Neighbor (k-NN) classification, which is a different approach to machine learning than SVMs are. Unlike SVMs, the goal with k-NN classification is not to find a lineair function or hyperplane to separate data points in a future space, but instead to look only at the k classes of instances that are within close distance of the new instance. This means that a new unknown instance is assigned the same class as the k number of instances around it.

Using the same data and features as with the SVM experiment, the experiment is conducted using TiMBL, this time using 'leave one out' for creation of test and training data. This is similar to tenfold cross-validation but instead of using folds of equal size, each instance is left out of the dataset to serve as testdata, while the learner is trained on the remaining examples. The advantage of using such methods is that they factor out the bias of choosing any set of instances from small datasets like the one in this experiment.

With an accuracy of 86.1%, precision of 81.9% and a recall of 80.5%, it can be concluded that there is nothing wrong with the chosen feature set, as was suggested by the poor performance of the SVM classifier. Although the baseline still outperforms this approach in terms of precision, accuracy is slightly better and the recall shows a dramatic improvement over the baseline, especially over the experiment using SVMs. Table 6 has an overview of the results of all three experiments.

	Accuracy	*Precision*	*Recall*
Baseline	85.0%	89.2%	68.1%
SVM	63.9%	76.7%	5.6%
TiMBL	86.1%	81.9%	80.5%

Table 6. Classification Results

4.2.4. Discussion

Based upon the first results with the SVM classifier compared to the results of the baseline classifier it can be said that the features used in the experiment are not better than using a simple bag-of-word method. After experimenting with another learning algorithm it can be concluded from the results

that the features used are useful after all, it just depends on the learning algorithm used. A dramatical increase in recall combined with a good score for both precision and accuracy justifies the use of the described feature set with a machine based learner like TiMBL as the weapon of choice in classifying concert reviews.

5. PRESENTATION

The final step in the development of the application is creating a user interface in which the results of the retrieval and classification task will be presented to users. Since the focus in this thesis is mainly on retrieval and classification, this section will only describe some ideas and propose a prototype for a possible interface. Evaluations of this interface are subject for later research.

Users should be able to get the user generated content related to a concert in two different ways. The first way is by browsing through lists that are sorted by date, venue, city or concert. An overview of all events that are related to the list view the user selected is given. Lists show all the available information on a particular event, e.g., when the list by city view is selected the user is presented with the date of an event, band name, venue name and the amount of UGC that has been found. It is possible for the user to sort that list on each information column, to facilitate finding what the user is looking for. This browsing option is ideal for people who want to find all the UGC related to an event, but don't know too many details about it anymore, or who just want to browse through the entire collection and read other people's opinions of different concerts.

Another option is by using a simple search option, which enables a user to search any of the information fields, or combinations of them. This way, a user who is looking for a particular event can quickly find the event he or she is looking for, without the need of scrolling through large lists of events.

Using either option, the user will ultimately be presented with the event overview page, which is an automatically generated page consisting of a top horizontal bar with three columns below. In the top horizontal bar all the information for the event is summarized, along with a couple of related browsing options, e.g., 'find more concerts by this band'. The three columns represent the three different information types that are available in the database: UGC on the event, UGC on other concerts by that band and UGC on the band in general. Each column has a different way of showing the information in it and the importance of each column should be visibly clarified, for instance using different widths for the columns or subtle color gestures. Figure 4 shows an example of this page.

The first column, on the far left, shows all the UGC related to the event. Because this information is the information the user is primarily looking for, it should be as extensive as possible. This means that it should show the entire review, or in case of long reviews the first part of it, including the title of the review at the top and a link to the original at the bottom of the review. Moreover, some metadata about the review, like the posting date, is shown.

Initially the reviews in this column are sorted by date, with the latest entry on top, but users can change this order by rating the reviews using a rating scheme like thumbs up, thumbs down. These ratings could eventually be used as additional learning material for the review classifier. Nevertheless, it should always be possible for the user to order the reviews in two possible ways: by rating and by date. In principle, the number of reviews in this column is unlimited, but when there are too many it should be possible that the reviews are divided over multiple pages, accessible to the user by tabs or a link.

Next to the event column, in the middle of the page, is the second column which shows other concert reviews. On a classification level it is hard to distinguish between concert reviews that belong to a certain event and those that do not. On the presentation level, this distinction is made based on the posting dates of the reviews. All those that are posted before the date of the event can never be reviews of the event and are therefore placed in this column. For reviews posted after the event date, this distinction is made based on the number of days that lie between the event and the posting date.

Because the reviews in this column are not the primary goal of the user's information needs, this column should show the reviews less detailed, i.e. only a snippet of the text around the band name in the review. Just as in the left column with the full reviews, a title and source are displayed, so the user can read the full review by clicking on the source. In contrast to the first column, ratings are not shown and it is also not possible to change the order of the reviews. Also, the number of results that is shown in the second column is fixed, so the page doesn't get too full with reviews. The column is ordered by date, showing the newest review first. In both review columns the name of the band is boldfaced, so the user has direct feedback that these reviews are in fact about the band he expects them to be.

The last column shows all weblogposts and newspaper articles that do not contain concert reviews. Because these posts are the least relevant to the user's information needs, they are shown with even less detail than in the second column: only the title of the article is shown and a user can click on this article to read the entire text. The entries in this column are ordered by date, newest first and a maximum number of entries, ten or twenty, is shown.

A possible extension to this layout is the inclusion of other types of UGC, like multimedia. These can be put in a new horizontal bar between the three columns and the top bar in which the details of the event are placed. Movies and pictures made by people who attended the concert are mixed and placed in the bar, with the newest entry on the left.

Fig. 4. Screenshot of the event overview page

6. CONCLUSION

In this thesis the development a system that aggregates user generated content related to pop concerts is discussed, using a three step approach. Although the system is based upon historical data from a limited window in time and tested using a limited source of user generated content, the results of this thesis can be useful in developing a real life application.

The results from the first step, retrieval of relevant data, show that using a query-based approach to retrieval can be fruitful, but depends heavily on the type of query that is used. Overall, phrase queries that are enriched with the name of the venue where the event took place return the most usable documents. This makes sense, as the name of the venue is usually a very unique name and by searching for a band-name as a phrase instead of a number keywords a lot of ambiguity errors can be prevented. While precision is high for these types of queries, the downside of this method lies in recall. Phrase queries are not very flexible, e.g., documents with varieties in the spelling of a band or venue name or typos are hard to find. Nevertheless, due to the high precision and average recall this method is preferred over the other query methods.

The results from the second step, classification of user generated content, show that using a naïve baseline of a bag-of-word method provides adequate results in classifying documents, but that these results can certainly be improved by using a more advanced feature set. Striking were the differences in the quality of classification while using two different machine learning algorithms on the same feature-set. The cause of the bad performance of the advanced feature set in combination with the SVM learning algorithm is not easy to point out, but a couple of factors can be of influence.

The first factor of influence can de derived from the frequently occurring words in both negative and positive instances of documents. There are few real significant words for either type of document, which leads to the conclusion that the documents are, on a term basis, very similar to each other, thus reducing the options of distinguishing between them successfully. A second factor can be the over-representation of negative instances in test and training data. In both the baseline and the experiment with the advanced feature-set with Support Vector Machines as learner, recall is low and a tendency to classify documents as negative instances can be observed.

These two factors can cause problems while using a linear classifier as SVM is, because it is very hard to compute an optimal hyperplane in a feature-space with so many datapoints of opposite polarities being close to each other, such that the margin of error is minimized. Such a context, combined with the over-representation of one type of instances results in a classifier favoring that instance, as can be seen in the results from both baseline and the first experiment. For the baseline this effect was less strong, due to the fact that the baseline used a much larger feature set which made it possible to make finer granulated distinctions between positive and negative instances.

Nevertheless, performance could be improved more. Us-

ing another type of machine learner, based on k-Nearest Neighbor algorithms caused recall to improve in particular. This can be explained from the fact that in nearest neighbour classification there is no need to find an optimal hyperplane which separates the datapoints in a certain feature space, but instead classification of an unknown instance is based upon the class of its neighbouring instance. In this way, even in a feature space with lot of instances of different polarities mixed together, classifications can be made pretty accurate, as the results from the experiment with the memory based TiMBL learner show.

With respect to the main research question of this thesis it can be concluded that pop concerts do have their effect in the creation of user generated content. The relatively large number of weblog posts discussing concerts in an unfocused dataset as the one used in the experiments proves this. Capturing this user generated content in a web environment can be done using a variety of methods for retrieval and classification. Simple querying with phrase queries and venue names proves to be a computationally inexpensive and yet effective way of retrieving user generated content. An advanced feature set consisting of a combination of bag-of-word methods, part-of-speech tagging and date properties of documents, used together with a memory based learning algorithm proves to perform pretty well in classifying concert reviews, with precision, recall and accuracy scores all above 80%. The results are presented in a accessible web interface, clearly showing the different types of information related to a concert to the user. Because differentiation between concert and event type of reviews can not be made by the classifier yet, this separation is made by looking at the posting date of a review with respect to the event date.

Finally, some remarks should be made about possible improvements for classification and retrieval. A possible improvement would be the use of n-grams instead of single word occurrences, since both negative and positive documents share a lot of words, and this might be different for n-grams. Further, expanding a query or feature-set with particular terms that are related to particular bands, like band-members, or song-names, can improve the quality of the results for both tasks. Another option is looking at a different dataset than the one used in this paper. Although the Verkiezingskijker data set is clearly usable, a corpus consisting of more specific concert reviews, like taken from sites on which fans discuss concerts, can improve the learning process by adding more positive instances to the training-data.

7. REFERENCES

[1] Gilad Mishne and Maarten de Rijke, "A study of blog search," in *Proceedings of the European Conference on Information Retrieval 2006*, 2006, pp. 289–301.

[2] Gilad Mishne and Nathalie Glance, "Predicting movie sales from blogger sentiment," *AAAI 2006 Spring Symposiumon Computational Approaches to Analysing Weblogs*, 2006.

[3] Gottfried Mayer-Kress and Cathleen Barczys, "The global brain as an emergent structure from the worldwide computing network, and its implications for modeling," *The Information Society*, vol. 11, pp. 1–27, 1995.

[4] Stephan Bauman and Oliver Hummel, "Using cultural metadata for artist recommendations," in *Proceedings of the WedelMusic Conference*, 2003.

[5] B. Whitman and S. Lawrence, "Inferring descriptions and similarity for music from community meta-data," in *Proceedings of the International Computer Music Conference*, 2002.

[6] X. Hu, J.S. Downie, K. West, and A. Ehmann, "Mining music reviews: Promising preliminary results," in *Proceedings of the 6th International Symposium on Music Information Retrieval*, 2005, pp. 536–539.

[7] D. Evans and C. Zhai, "Noun-phrase analysis in unrestricted text for information retrieval," in *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, 1996, pp. 17–24.

[8] P. Knees, E. Pampalk, and G. Widmer, "Artist classification with web-based data," in *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR)*, 2004, pp. 517–524.

[9] Valentin Jijkoun, Maarten Marx, Maarten de Rijke, and Frank van Waveren, "Electoral search using the verkiezingskijker: an experience report," in *WWW '07: Proceedings of the 16th international conference on World Wide Web*, New York, NY, USA, 2007, pp. 1155–1156, ACM Press.

[10] T. Joachims, *Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning*, MIT-Press, 1999.

[11] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999.

[12] Erik F. Tjong Kim Sang, "Generating subtitles from linguistically annotated text," *Internal report Atranos project, WP4-12, University of Antwerp*, 2003.

[13] Gilad Mishne, *Applied Text Analytics for Blogs*, Ph.D. thesis, University of Amsterdam, 2007.

[14] Walter Daelemans and Antal Van den Bosch, *Memory-Based Language Processing*, Cambridge University Press, 2005.