# Identifying Breakpoints in Public Opinion

Cuneyt Gurcan Akcora, Murat Ali Bayir, Murat Demirbas
Computer Science & Eng. Department
University at Buffalo, SUNY
14260, Buffalo, NY, USA
{cgakcora, mbayir, demirbas}@cse.buffalo.edu

Hakan Ferhatosmanoglu
Computer Science & Eng. Department
The Ohio State University
Columbus, OH 43210, USA
hakan@cse.ohio-state.edu

## ABSTRACT

While polls are traditionally used for observing public opinion, they provide a point snapshot, not a continuum. We consider the problem of identifying breakpoints in public opinion, and propose using micro-blogging sites to capture trends in public opinion. We develop methods to detect changes in public opinion, and find events that cause these changes.

Our experiments show that the proposed methods are able to determine changes in public opinion and extract the major news about the events effectively. We also deploy an application where users can view the important news stories for a continuing event and find the related articles on web.

## Categories and Subject Descriptors

H.2.8 [**Information Systems**]: Database Management—*Database Applications, Data Mining*; H.3 [**Information Systems**]: Information Storage and Retrieval

## General Terms

Opinion Mining, Emotion Corpus, Microblogging, Sentiment Analysis.

## 1. INTRODUCTION

Since 1824 [1], polls have been used to take a snapshot of public opinion, but they cannot reach many people nor capture opinions about the topics that are not asked in the questionnaire. Moreover, while events unfold rapidly and public opinion changes with those events, polls cannot account for the temporal changes in public opinion. With the advance of micro-blogging sites like Twitter [7, 10], we are now able to observe individual opinions and keep up with the changes in the public opinion. When carefully aggregated and classified, individual opinions can give us a better understanding of how some events are received by the public.

[1]Conducted in the contest for the United States presidency.


*1st Workshop on Social Media Analytics (SOMA '10),* July 25, 2010, Washington, DC, USA.


In this paper, we propose efficient methods to identify and classify opinions in a large stream of information, and pinpoint related events that stimulate users to express their opinions.

In particular, the contributions of this paper are as follows:

- We develop and utilize an emotion corpus to detect emotions in tweets. This method enables expanding opinion representation from binary options ("positive or negative") to multiple dimensions by providing more granularity in classification.
- We propose combining set and vector space models to observe the public opinion and detect changes over time. From the experimental results, we found that using these two methods together eliminates false positives and improves the accuracy of our findings.
- We develop a dynamic scoring function to give a synopsis of news (in terms of prominent words) that led to breakpoints in public opinion.
- We create a customized event tracking application that can notify users without flooding them with every new entry about the event. We show that our application is more user friendly than the Google Alert[2] service.

## 2. RELATED WORK

Opinion Mining has received great attention recently and researchers started to investigate people's opinion about certain topics or news [6].

Existing opinion mining methods are usually grouped under two categories [8, 11] called document based and attribute based approaches. These approaches are focused on characterizing user opinions as positive or negative over domain specific web sites [4, 13] for different applications.

As a document level approach, Turney et al. [14] proposed determining polarity of documents by using semantic orientation of extracted phrases. As an example of attribute based approaches, Zhuang et al. [15] proposed a method for grouping movie reviews based on frequent opinion terms. Differing from these supervised approaches, we propose using a finer granularity classification (8 emotion classes) for opinions.

To account for the temporal changes in public opinion, a related work to our approach is proposed by Ku et al. [9]

[2]http://www.google.com/alerts

where the authors used the language characteristics of Chinese. In temporal dimension, their method captures opinions and shows changes in overall sentiment about candidates in a presidential election.

# 3. METHODOLOGY

We begin our discussion for methodology by first explaining what indicates a change in public opinion in streaming tweets. For this purpose, we note two observations on Twitter data.

**Observation 1:** If an event results in a change of public opinion, more tweets contain emotion words. Furthermore, **emotion pattern** of tweets in that time period is different from the emotion pattern of the preceding period, but more similar to the emotion pattern of tweets in the following period, i.e, the news has an enduring impression on public.

- **Example Tweet:** (Transgression claims admitted by Woods.) *Tiger Woods - What a disappointment.*

**Observation 2:** If an important story about the event appears, the **word pattern** of tweets is different from last period. On the other hand, the same word pattern repeats in the next period, i.e, tweets contain similar words in the next period as still the same topic is discussed.

- **Example Tweet:** (Companies start ending sponsorship agreements.) *Accenture Dumps Tiger Woods From Corporate Homepage.*

Following these observations, we conclude that, to claim a change in the public opinion, the **emotion pattern** and the **word pattern** must change according to these observations. We are looking for news that are both major events and opinion changers. In Section 3.1 we discuss how we find emotion and word patterns and use mentioned observations to detect opinion changes. We continue with finding topics related to the events in section 3.2

## 3.1 Opinion Detection

For the emotion pattern, we use an emotion corpus based method, while using set space model for the word pattern.

**Emotion Corpus Based Method** is based on vector space model for calculating document similarity. For the emotion detection in tweets, we use an emotion corpus that is based on 8 basic classes, E={**Anger, Sadness, Love, Fear, Disgust, Shame, Joy, Surprise**}, from [12]. We built a 309 word emotion corpus to populate those 8 classes. Each class represents a dimension in the Boolean emotion vector of a tweet. We look for emotion words in a tweet, and if found, set the corresponding class dimension in the emotion vector to 1, otherwise it remains 0.

- **Tweet:** *I was on main street in Norfolk when I heard about tiger woods updates and it made me feel angry,* on 2009-12-11. **Emotion vector:** $(1, 0, 0, 0, 0, 0, 0, 0)$.

For all the tweets in a chosen time interval, a centroid of all corresponding emotion vector dimensions is calculated, and this centroid is considered a document for each interval.

For a given time interval T that contains N tweets, let $V=\{v_1, v_2, \ldots, v_N\}$ be a set of vectors (with $l = 8$ dimensions each) generated from these tweets. We define centroid $\bar{v}$ for period $T$ as:

$$\bar{v} = (\frac{\sum_{k=1}^{k=N} v_k^1}{N}, \frac{\sum_{k=1}^{k=N} v_k^2}{N} ..., \frac{\sum_{k=1}^{k=N} v_k^l}{N}) \quad (1)$$

After finding centroid vector for each interval, we define the opinion similarity between two intervals $T_1$ and $T_2$ by calculating cosine similarity between their centroid vectors:

$$Sim(T_1, T_2) = \frac{\bar{v}_1.\bar{v}_2}{|\bar{v}_1||\bar{v}_2|} \quad (2)$$

**Set Space Model** prescribes representing each interval by a single document which is the union of the tweets posted in that particular time interval. After removing the stop words and stemming the terms using Porter stemmer [3], we collect all terms in a hash set for each interval. We define the similarity between two intervals $T_1$ and $T_2$ by calculating Jaccard Similarity [2]:

$$Sim(T_1, T_2) = \frac{|(Set)T_1 \cap (Set)T_2|}{|(Set)T_1 \cup (Set)T_2|} \quad (3)$$

To find the changes, neither corpus based method nor the set space model alone is suitable. For the corpus based method, a change in the centroid can be misleading when the interval has very few emotion words compared to its neighbors. For the set space model, a change in similarity does not by itself imply an opinion change, because not all of the words are emotion words. In our method, we first analyze vector space similarity. If we detect a possible change, we validate it by analyzing the Jaccard Similarity. Following the observations 1 and 2, if both methods detect the change, we report that point as a breakpoint.

$T_n$ is a time break, if the followings are satisfied in both corpus based method and set space model:

$$Sim(T_{n-1}, T_n) < Sim(T_{n-2}, T_{n-1}) \quad (4)$$

$$Sim(T_{n-1}, T_n) < Sim(T_n, T_{n+1}) \quad (5)$$

## 3.2 Breakpoint Representation

After detecting the changes, we set out to identify the events that caused these changes. To this end, we look for the prominent words of an interval to represent the breakpoint. For the prominent word selection, we propose a TfIdf based dynamic scoring function. The algorithm should effectively find recently emerging keywords to guide users into catching breaking news and pay special attention to the words which emerge in a period and start appearing in more periods as time progresses.

**The Streaming TfIdf Algorithm**. To identify the events that caused breakpoints, we need to find keywords that represent the topics of these events. We propose the Streaming TfIdf algorithm for extracting event related keywords from an information stream of tweets.

**Document Phase**. For breakpoint representation, the same time interval length in the opinion detection is used, and for every time interval $T_n$, a document $D_n$ contains the union of stemmed words from all tweets in that period. For word $x$ in document $D_n$, Term Frequency $Tf_{x,D_n}$ is

[3]http://tartarus.org/martin/PorterStemmer/

calculated as:

$$Tf_{x,D_n} = \frac{Count_{x,D_n}}{\sum_{k=1}^{n} Count_{k,D_n}} \quad (6)$$

For the total count of documents up to document $D_n$, Inverse Document Frequency of a word x in document $D_n$, $Idf_{x,D_n}$ is calculated as:

$$Idf_{x,D_n} = \log(\frac{n}{|\{\forall k, k \leq n : x \in D_k\}|}) \quad (7)$$

Note that, $n$ is not a fixed value. As we move from the oldest document to the newest document, the total number of documents, $n$, increases. By this parameter, the first appearance of a keyword will always have a bigger Idf value, and the following appearances of the word will have smaller values.

Based on the calculated $Idf_{x,D_n}$ and $Tf_{x,D_n}$, we calculate the $TfIdf$ value as:

$$TfIdf_{x,D_n} = Tf_{x,D_n} \times Idf_{x,D_n} \quad (8)$$

**Prominence Update Phase**. For a keyword $x$ that recently appeared in $D_n$, we define the $Tf_{x,D_o}$ for the word $x$ in document $D_o$ where $o < n$ as:

$$tf_{x,D_o} = tf_{x,D_o} + F(T_o, T_n) \times tf_{x,D_n} \quad (9)$$

Here, we apply a decay function $F(o, n)$ to prevent the word $x$ in the document $D_n$ to increase the $Tf$ value of $x$ in a too old document $D_o$. This follows from the fact that, tweets are highly temporal, i.e, new events tend to affect user tweets only for a short period of time. As we move forward in the time domain, a keyword in a new period should not increase the prominence of a keyword in a way back period, because it is highly unlikely that appearance of a keyword is because of a very old event.

For the period numbers $o$ and $n$, we define the decay function for two periods $T_o$ and $T_n$ as:

$$F(T_o, T_n) = 1/(n - o) \quad (10)$$

For the updated $Tf$ values of the keyword $x$ in document $D_o$, we re-calculate the $TfIdf_{x,D_o}$ as:

$$TfIdf_{x,D_o} = Tf_{x,D_o} \times Idf_{x,D_o} \quad (11)$$

We choose $p$ words with highest $TfIdf$ values from each document, and call them prominent words of that document.

# 4. EXPERIMENTAL RESULTS

In this section, we present experimental results of our methods on Twitter. We analyzed data about two topics, (1)Fort Hood shootings in Texas, USA, November 05, 2009 and (2)Tiger Woods, November 27, 2009 car accident. Due to space limitations here we only present the Tiger Woods news story. We used a Twitter search engine, Twopular [4] to collect data. We processed 258548 tweets, and found 23280 emotion words in those tweets. Figure 1 shows the tweet count of each day.

[4]www.twopular.com

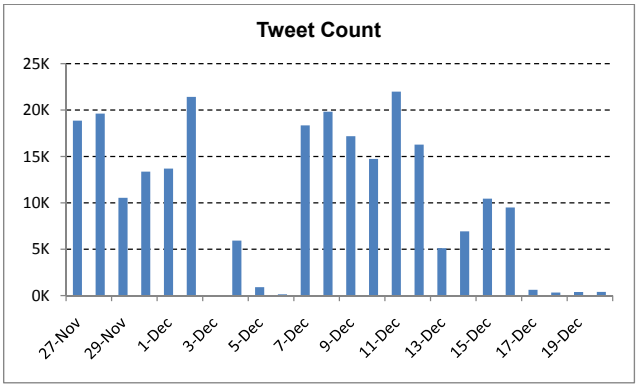


**Figure 1: Tweet Count of Days**

## 4.1 Opinion Detection

The length of time intervals is an important factor in our analysis. We evaluated unit lengths varying from 2 hours to 24 hours. Intervals shorter than 12 hours lead to biased results, because they contain too few tweets to form a meaningful sample. On the other hand, intervals longer than 24 hours are not suitable for the problem domain (media news cycle). We chose 12 hours, because it is the shortest interval to provide meaningful data besides enabling us to capture events in fine granularity.

In our data for 20 days, we found 8 possible breaks by Emotion Corpus Method (Figure 2) {**5, 10, 17, 23, 25, 27, 32, 36**}, and 5 of them {**5, 10, 23, 25, 27**} were also captured by Jaccard similarity (Figure 3). Figure 2 contains black bars that represent outlier intervals with very few tweets.

We tested our findings with a time line of Tiger Woods related events from CNN, ABCNews and ESPN [5]. Our 3 validated breaking points are related to the following events in successive order: (5)Transgression claims accepted by Tiger Woods, (10)more women alleged to have affairs with Woods, (23) Gatorade ends a sponsorship agreement with Woods, and Twitter users start writing thousands of jokes about Woods with Santa Claus #hashtags nearing Christmas. Among the validated breakpoints, **25** and **27** are false positives.

## 4.2 Breakpoint Representation

Upon detecting opinion changes in the Tiger Woods case, we found frequent keywords of all periods, and by using the Streaming TfIdf algorithm, we extracted the prominent words from these keywords.

While creating documents for each 12 hour period, we put top $F$ most frequent words into their respective documents. During this process, we used the Porter Stemmer to remove the commoner morphological and inflexional endings of words and analyzed the frequency distribution graph of the words. We found 50 to be the best choice because for values larger than 50, big clusters of words with low frequencies appear.

For the number of **prominent words** $p$, we used $p = 5$. The first document has the prominent words: **crash, report, florida, injur, golfer**. The prominent words can many times be self explanatory: **accenture, drop, stop,**

[5]http://sports.espn.go.com/golf/news/story?id=4922436

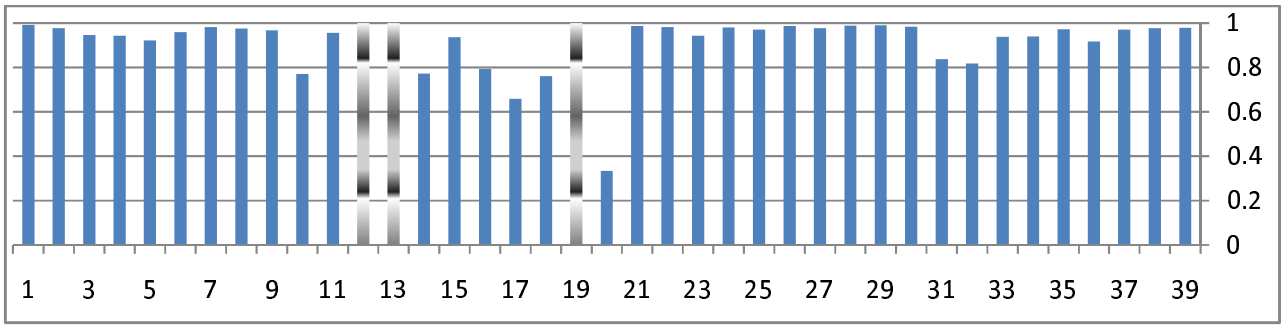


**Figure 2: Emotion Vector Similarity of two successive intervals**

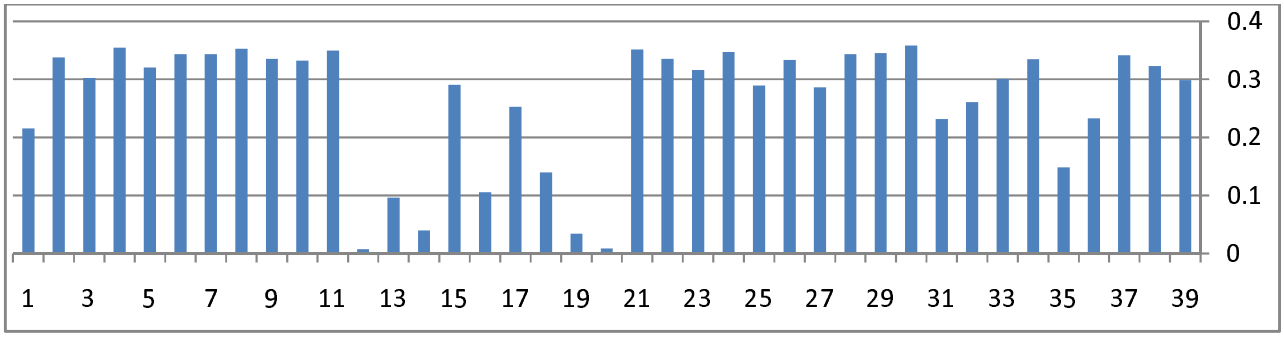


**Figure 3: Jaccard Similarity of two successive intervals**

**golfer, sponsorship**. This refers to the Accenture's decision to drop a sponsorship with Tiger Woods. The algorithm can successfully detect appearance dates of emerging topics. While prominent words of the 11th document with the traditional $TfIdf$ does not include the word "voicemail", the Streaming TfIdf algorithm correctly identifies it as breaking news and adds it to the prominent words.

Apart from identifying the prominent words, the algorithm correctly discriminates against words that are not related to the events. In the $11th$ interval, the word "Afghanistan" is in the set of prominent words. It is because of the tweets that protest Tiger Wood headlines while "Afghanistan war" gets more violent. In the following days, the prominent word set of the document is updated and "Afghanistan" disappears from the prominent word set, as it is not actually related to the event.

The breakpoint representation method identifies the significant periods as 6, 11 and 24. Note that, a break on the $(n)th$ bar in the similarity graphs (Figures 2- 3) indicates an opinion change between $(n)th$ and $(n+1)th$ time periods. For these breakpoints, Table 1 gives us the prominent words for $(n+1)th$ intervals.

**Run Time Analysis** of our methods show a linear characteristic as the tweet count increases. In order to test scalability, we experimented with $5000, 10000$ and $20000$ tweets and found the run time of our methods to be $24224, 45985$ and $92867$ miliseconds on AMD Turion Dual-Core 2.00GHz processor.

| Period | Prominent Words |
|---|---|
| 1 | ***crash, florida, injur, golf, accident*** |
| 6 | ***crash, wife, accident, mistress, golf*** |
| 11 | ***voicemail, wife, f***, golf, cheat*** |
| 24 | ***drop, stop, santa, claus, gatorade*** |

**Table 1: Prominent Values for Significant Periods**

## 5. CUSTOMIZED NEWS TRACKING

We developed a news tracking application on Twitter. The resulting application can be seen at the project web site[6], and its screenshot is given in Figure 4. The application uses an interactive Javascript interface that lists the tweet counts of each period. The user can click on the period columns to see the events of a time period depending on the prominent words. For each period, we search for the articles that are published in the date range of the period. We are not storing those web links in a database, because the links can be removed or re-located over time.

Google Alert offers such a customized web service, and it provides a system which notifies users by email when a chosen keyword has a new entry on web. Whereas Google sends updates about every entry on a tracked keyword, our application observes the public opinion to identify breaking points and finds keywords of important events to notify users about them.

## 6. CONCLUSIONS

In this paper we presented an efficient way to observe public opinion on temporal dimension. Our methods can iden-

[6]http://ubicomp.cse.buffalo.edu/upinion/

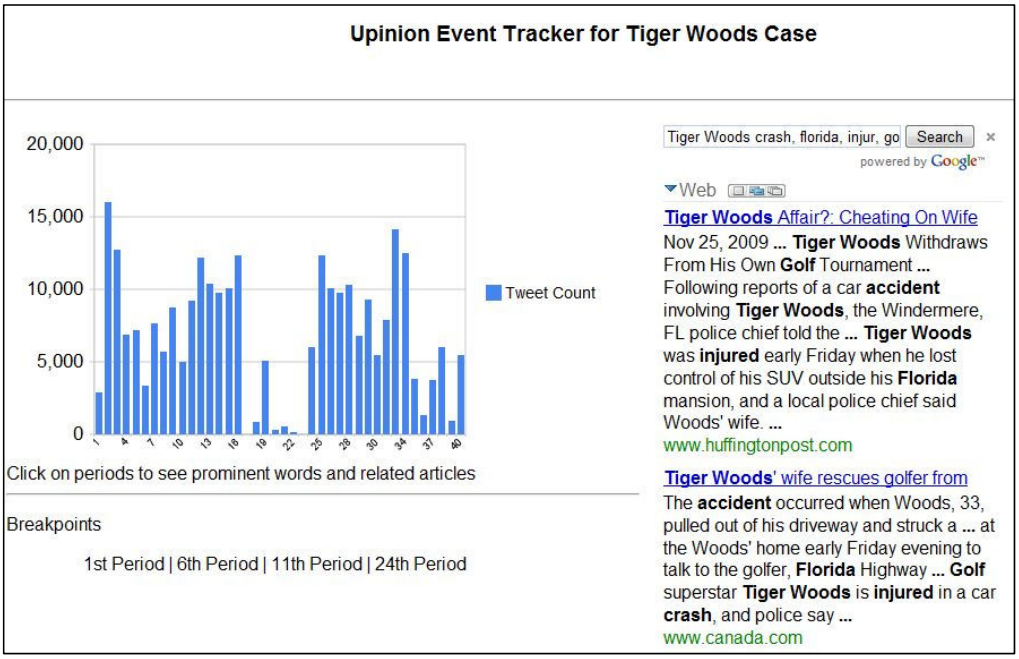


**Figure 4: Upinion Application**

tify break points, and find related events that caused these opinion changes. We tested our results with the timeline of Tiger Woods case and showed the accuracy of our results. We developed an application that can serve users with news pages depending on the time period. We are currently working on expanding the emotion corpus for eliminating outlier intervals in our analysis.

As a future work, we are planning to develop customized version of our web service that enables web users to track their selected topics on Twitter. We are also working on distributed implementation of our system over Hadoop [7] Map/Reduce framework. Map/Reduce [5] allows large software frameworks [1, 3] to process unlimited amount of data in a distributed manner. By using power of Map/Reduce paradigm, we are planning to handle millions of tweet at the same time belonging to multiple topics.

## 7. REFERENCES


[1] M. A. Bayir, I. H. Toroslu, A. Cosar, and G. Fidan. Smart miner: a new framework for mining large scale web usage data. In *WWW*, pages 161–170, 2009.

[2] M. W. Berry, editor. *Survey of text mining: clustering, classification, and retrieval*. Springer, 2004.

[3] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li. Context-aware query suggestion by mining click-through and session data. In *KDD*, pages 875–883, 2008.

[4] K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *WWW*, pages 519–528, 2003.

[5] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. In *OSDI*, pages 137–150, 2004.


[7] http://hadoop.apache.org/


[6] N. Diakopoulos and D. A. Shamma. Characterizing debate performance via aggregated twitter sentiment. In *Conference on Human Factors in Computing Systems (CHI)*, April 2010.

[7] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65. ACM, 2007.

[8] W. Jin, H. H. Ho, and R. K. Srihari. Opinionminer: a novel machine learning system for web opinion mining and extraction. In *KDD*, pages 1195–1204, 2009.

[9] L. Ku, Y. Liang, and H. Chen. Opinion extraction, summarization and tracking in news and blog corpora. In *Proceedings of AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs*, pages 100–107, 2006.

[10] H. Kwak, C. Lee, H. Park, and S. B. Moon. What is twitter, a social network or a news media? In *WWW*, pages 591–600, 2010.

[11] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, 2007.

[12] W. G. Parrott, editor. *Emotions in social psychology: essential readings*. Psychology Press, 2001.

[13] A. Popescu and O. Etzioni. Extracting product features and opinions from reviews. In *EMNLP-05*, 2005.

[14] P. D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424, 2002.

[15] L. Zhuang, F. Jing, X.-Y. Zhu, and L. Zhang. Movie review mining and summarization. In *CIKM-06*, 2006.