# Opinion mining

# Introduction – facts and opinions

- Two main types of textual information on the Web.
  - Facts and Opinions
- Current search engines search for facts (assume they are true)
  - Facts can be expressed with topic keywords.
- Search engines do not search for opinions
  - Opinions are hard to express with a few keywords
    - What do people think of Motorola Cell phones?
  - Current search ranking strategy is not appropriate for opinion retrieval/search.

# Opinions are user-generated content

- Word-of-mouth on the Web
  - One can express personal experiences and opinions on almost anything, at review sites, forums, discussion groups, blogs ... (called the user generated content.)
  - They contain valuable information
  - Web/global scale!!
  - Our interest: to mine opinions expressed in the user-generated content
  - A very challenging problem.
  - Practically very useful.

# Applications

- Businesses and organizations: product and service benchmarking. Market intelligence.
  - Business spends a huge amount of money to find consumer sentiments and opinions.
  - Consultants, surveys and focused groups, etc
- Individuals: interested in other's opinions when
  - Purchasing a product or using a service,
  - Finding opinions on political topics,
- Ads placements: Placing ads in the user-generated content
  - Place an ad when one praises a product.
  - Place an ad from a competitor if one criticizes a product.
- Opinion retrieval/search: providing general search for opinions
  - Predicting behaviours and trends in finance, medicine, politics

# Impact

- 81% of Internet users have done online research on a product 20% do so on a typical day
- Among readers of online reviews between 73% and 87% report that reviews had a significant influence on their purchase
- Consumers report being willing to pay from 20% to 99% more for a 5-star-rated item than a 4-star-rated item (the variance stems from what type of item or service is considered);
- 32% have provided a rating on a product, service, or person via an online ratings system, and 30% have posted an online comment or review regarding a product or service.

# A formalization of the opinion mining task

- Basic components of an opinion:
  - Opinion holder: The person or organization that holds a specific opinion on a particular object.
  - Object: on which an opinion is expressed (it can be described by features, e.g. for an hotel room: dimension, clean, silent, cost,..)
  - Opinion: a view, attitude, or appraisal on an object (or object feature) from an opinion holder.

# Opinion mining "grain"

- At the document (or review) level:
  - Task: sentiment classification of reviews
  - Classes: positive, negative, and neutral
  - Assumption: each document (or review) focuses on a single object (not true in many discussion posts) and contains opinion from a single opinion holder.
  - Example: Movie reviews
- At the sentence level:
  - Task 1: identifying subjective/opinionated sentences
    - Classes: objective and subjective (opinionated)
  - Task 2: sentiment classification of sentences
    - Classes: positive, negative and neutral.
    - Assumption: **a sentence contains only one opinion**; not true in many cases.
    - Then we can also consider clauses or phrases.
- Example: hotel reviews

# Opinion Mining Tasks (cont.)

- At the feature level (Example: product reviews, usually you want to know opinions on various features of the product to improve or to compare)
  - *Task 1: Identify and extract object features that have been commented on by an opinion holder*
  - *Task 2: Determine whether the* ***opinions on the features*** *are* positive, negative or neutral.
  - *Task 3: Group feature synonyms.*
- Opinion holders: identify holders is also useful, e.g., in news articles, etc, but they are usually known in the user generated content, i.e., **authors of the posts**.

# Feature-Based Opinion Summary

*"I bought an iPhone a few days ago. It was such a nice phone. The touch screen was really cool. The voice quality was clear too. Although the battery life was not long, that is ok for me. However, my mother was mad with me as I did not tell her before I bought the phone. She also thought the phone was too expensive, and wanted me to return it to the shop. …"*

….

**Feature-Based Summary:**

**Feature1: Touch screen**
Positive: 212

- *The touch screen was really cool.*
- *The touch screen was so easy to use and can do amazing things.*

**…**

Negative: 6

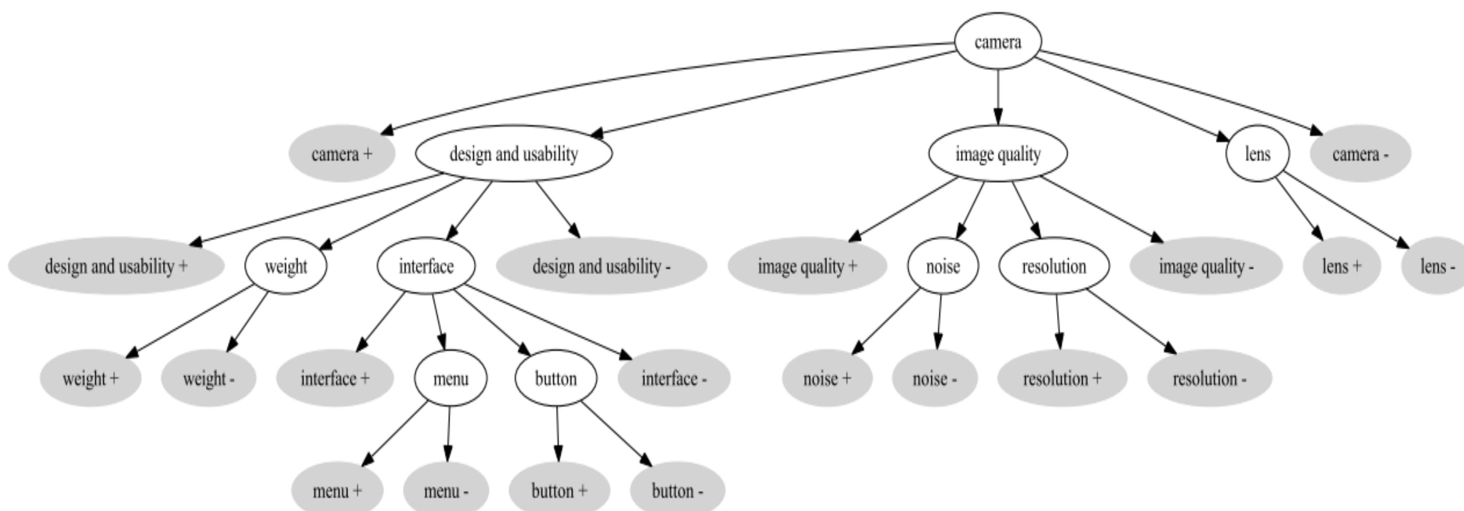
- The screen is easily scratched.
- I have a lot of difficulty in removing finger marks from the touch screen.

**…**

**Feature2: battery life**

**…**

# Needs “knowledge” to represent object features



# Opinion Analysis: Methods

- Two approaches to the problem:
  1. Machine-Learning (ML) solutions
  2. Lexicon-based solutions
  3. Hybrid solutions

- Each has advantages and disadvantages…

# Machine-Learning (ML) solutions

- 'Learn by example' paradigm
  - Provide an algorithm with lots of examples
    - Documents that have been *manually/semi-automatically* annotated with a category
      - *Supervised* learning
      - In our case: e.g., positive/negative reviews (e.g. Tripadvisor)
  - Algorithm extracts characteristic patterns for each category and builds a *predictive model*
  - Apply *model* to new text -> get prediction

- Things to note:
  - Typical machine-learning algorithms are typically used
    - SVMs, Naïve Bayes, ..
  - Focus is mostly on better *modelling* the documents -> design better features!
    - Enhance/replace standard bag-of-words approach

# ML for document classification

- Bag-of-words document representation: document → vector (“opinion” words can be considered, or, any word)
  - Example:

    $d_1$=“good…. average… excellent.. good..”

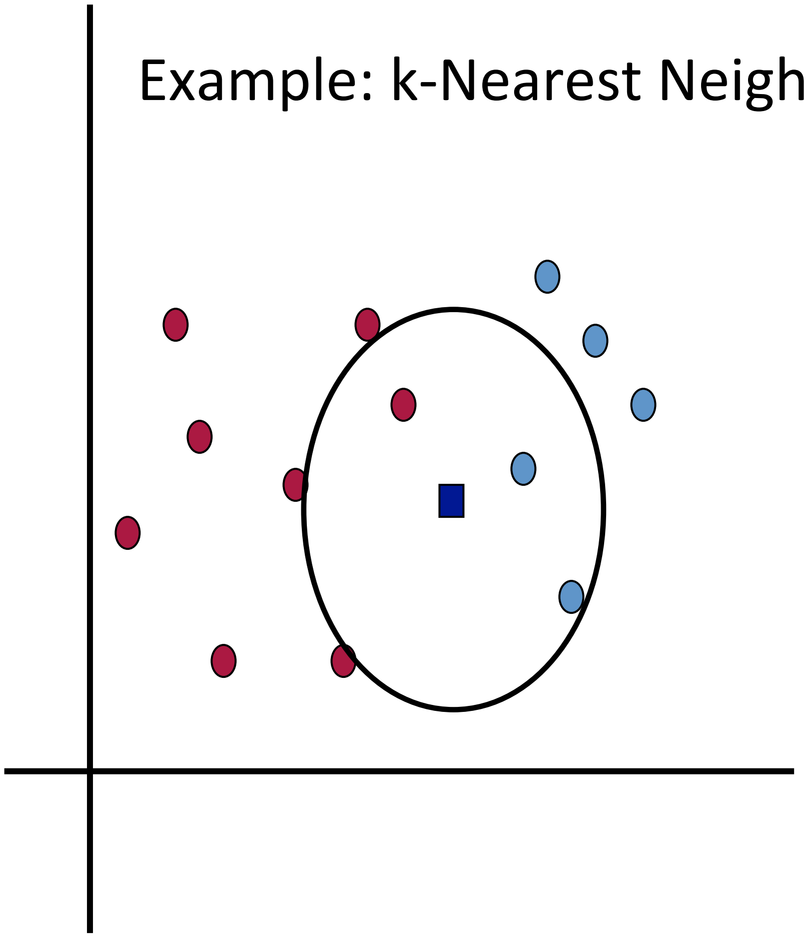
    $d_2$=“okay ..good.. average.. fine..”

    $d_3$=“good… okay.. okay…”
  - Then **Vocabulary**={“good”, “average”, “excellent”, “fine”, “okay”} and $d_1$ will be represented as:
    - **$d_1$={2,1,1,0,0}** if features are frequency-based or
    - **$d_1$={1,1,1,0,0}** if boolean-based

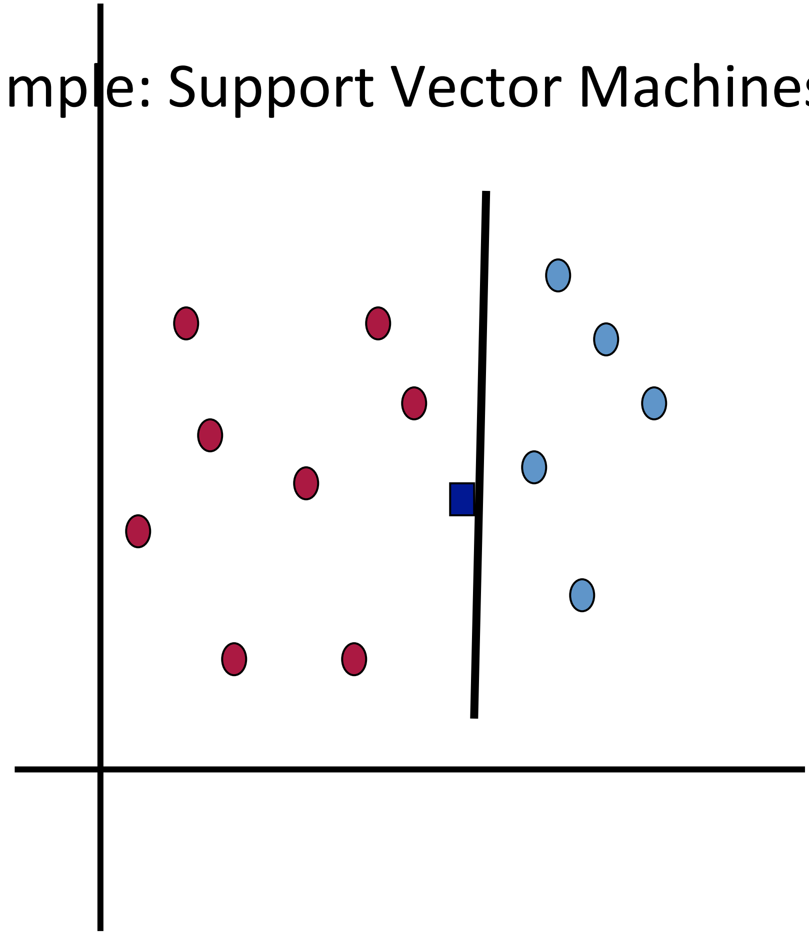
# Documents in a Vector Space - Classification

# Documents in a Vector Space - Classification

Example: k-Nearest Neighbours



Example: Support Vector Machines



# Machine-Learning solutions

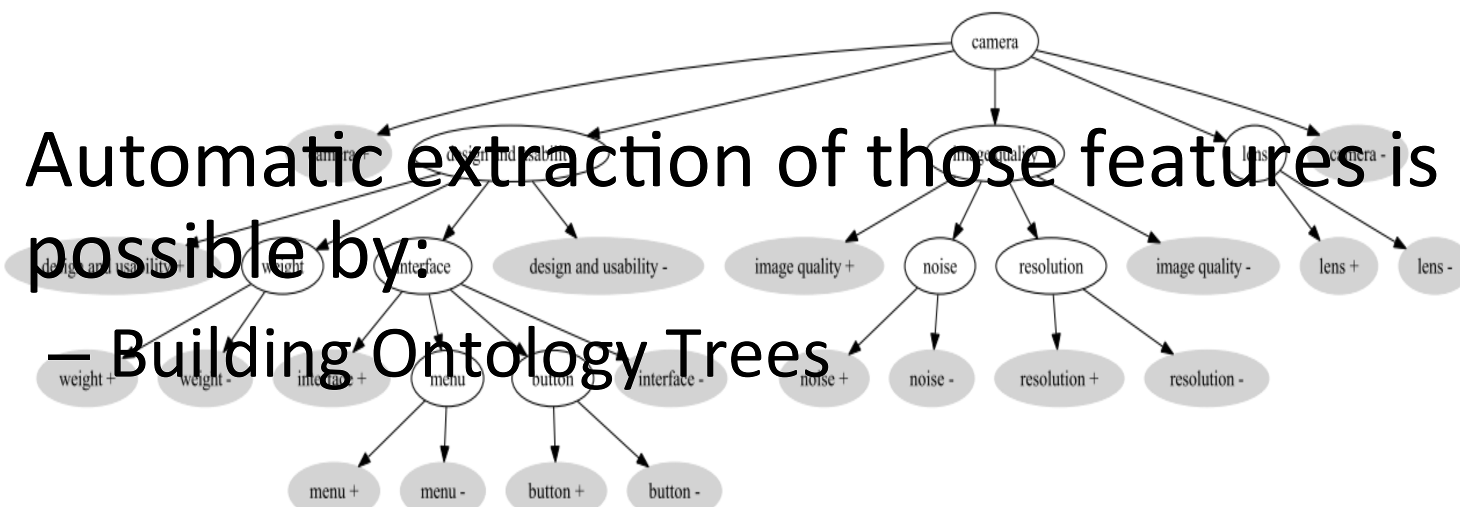
- Basic approach:
  1. Get manually annotated documents from the domain you are interested in.
     - e.g., positive and negative reviews of electronics products
     - This will be your training corpus
  2. Train any standard classifier using bag-of-words as features
     - Typical classifiers: Support Vector Machines (SVMs), Naïve Bayes
     - Naïve Bayes are super-easy to implement from scratch
     - Use boolean features not frequency-based
  3. Apply trained classifier to test corpus or application
- If you want to predict a rating, e.g., 1-5 stars (like in Tripadvisor): same as above, but use multi-class classification or regression:
  - Linear Regression, Support Vector Regression

# Machine-Learning solutions

- Typical extensions, focus on *extending/enhancing* the document representation. Instead of/in addition to bag-of-words features, ca use:
  - Extra features for *emphasised words , special symbols*
    - *wooooow*
    - *exlamations: !!!! ??*
    - *emoticons*
  - Higher order *n-grams* (e.g., bi-grams or bi-words)
    - "The movie was not very good, actually"
    - "The_movie / movie_was/ was_not/ not_very/ very_good / good_actually."
    - Helps capture features like: was_not (*negation)*, very_good (*intensifiers)*
  - Part-of-speech (*pos*) tags
    - "This is a love movie."
    - "This_DT is_VBZ a_DT love_NN movie_NN." (DT=determiner NN=noun)
    - Why? **Often adjectives are relevant for opinions**

# Feature-based Opinion Analysis

- As discussed, often the *Opinion Object* has different features
  - e.g., camera: lens, quality, weight.
- Often, such an aspect-based analysis is more valuable than a general +/-




- Automatic extraction of those features is possible by:
  - Building Ontology Trees

# Pros/Cons of the approach

- Advantages:
  - Tend to attain good predictive accuracy
    - Assuming you avoid the typical ML mishaps (e.g., over/unde
- Disadvantages:
  - Need for training corpus
    - Solution: automated extraction (e.g., Amazon reviews, Rotten Tomatoes) or crowdsourcing the annotation process (e.g., Mechanical Turk)
  - Domain sensitivity
    - Trained models are well-fitted to particular product category (e.g., electronics) **but underperform if applied to other categories** (e.g., turism)
    - Solution: train a lot of domain-specific models or apply *domain-adaptation* techniques
    - Particularly for Opinion Retrieval, you'll also need to identify the domain of the query!
  - Often difficult/impossible to rationalise prediction output

**Example: "small" is positive for a camera, negative for an hotel room**

# Sentiment-Annotated corpora

- http://www.**cyberemotions**.eu/data.html
- http://www.di.unito.it/~tutreeb/sentiTUT.html (in italian)
- Stanford Twitter Corpus: http://help.sentiment140.com/for-students
- HCR and OMD datasets: https://bitbucket.org/speriosu/updown
- Sentiment Strength Corpora: http://sentistrength.wlv.ac.uk/
- Sanders: http://www.sananalytics.com/lab/twitter-sentiment/
- SemEval: http://www.cs.york.ac.uk/semeval-2013/task2/

# Lexicon-based solutions

- Detect/extract the polarity of opinions, based on affective dictionaries
- Word-lists where each token is annotated with an 'emotional' value
    - e.g., positive/negative words or *words that express anger, fear, happiness, etc.*

  **Examples of affective dictionaries follow…**

- Add syntactic and prose rules to estimate the overall polarity of text:
  – Negation detection: "the movie wasn't good"
  – Exclamation detection: "great show!!"
  – Emoticon detection: "went to the movies ☺"
  – Emphasis detection: "You are gooooood"
  – Intensifier, diminisher word detection: "Very good movie" vs. "good movie"

# (Basic) lexicon-based approach

1. Detect emotion in two independent dimensions (numbers are weights of positive/negative opinionated words):
   1. Positive: $D_{pos}$: {1, 2,… 5}
   2. Negative: $D_{neg}$: {-5, -4,… -1}
2. (optional) Predict overall polarity by comparing them :
   – If $D_{pos} > |D_{neg}|$ then positive

   Example: "He is brilliant but boring"
   – Emotion('brilliant')=+3
   – Emotion('boring')=-2

   $D_{pos}$ =+3, $D_{neg}$=-2 => *positive*
3. Negation detection: "He isn't brilliant and he is boring"
   – *Emotion(NOT 'brilliant') = -2*
   – *Decreased by 1 and sign reversed*
4. *Exclamation detection: "He is brilliant but boring!!"*
   – *Increase weight of emphasized words*
   – *'boring'=-3*

# Extensions

- Of course, this is a *very* simplified description of methodology
- Typical extensions include:
  - Ability to *optimize* affective lexicon
    - Add / remove words (e.g. “small” is ok for a camera, is bad for an hotel room)
    - Manipulate affective weight based on training data
  - Proper syntax analysis
    - To locate the interdependencies between affective words and modifiers (“*It is barely appropriate*”)
  - Detection of user-defined keywords and their relation to affective text spans:
    - “went there, lol”
- Demo:
  - SentiStrength: http://sentistrength.wlv.ac.uk/
  - TweetMiner: http://mi-linux.wlv.ac.uk/~0920433/project/tweetmining.html

# Pros/Cons of the approach

- Advantages:
  - Can be fairly accurate independent of environment
  - No need for training corpus
  - Can be easily extended to new domains with additional affective words
    - e.g., “amazeballs”
  - Can be easy to rationalise prediction output
  - More often used in Opinion Retrieval
- Disadvantages:
  - Compared to a well-trained, in-domain ML model they typically underperform
  - Sensitive to affective dictionary coverage

# Affective Lexicons

- They have been extensively used in the field either for lexicon-based approaches or in machine-learning solutions
  - Additional features
  - Bootstrapping: unsupervised solutions (see previous)
- Can be created manually, automatically or semi-automatically
- Can be domain-dependent or independent
- A lot of them are already available:
  - Manual
    - LIWC: Linguistic Inquiry and Word Count
    - ANEW: Affective norms for English words
  - Automatic:
    - WordNet-Affect
    - SentiWordNet …

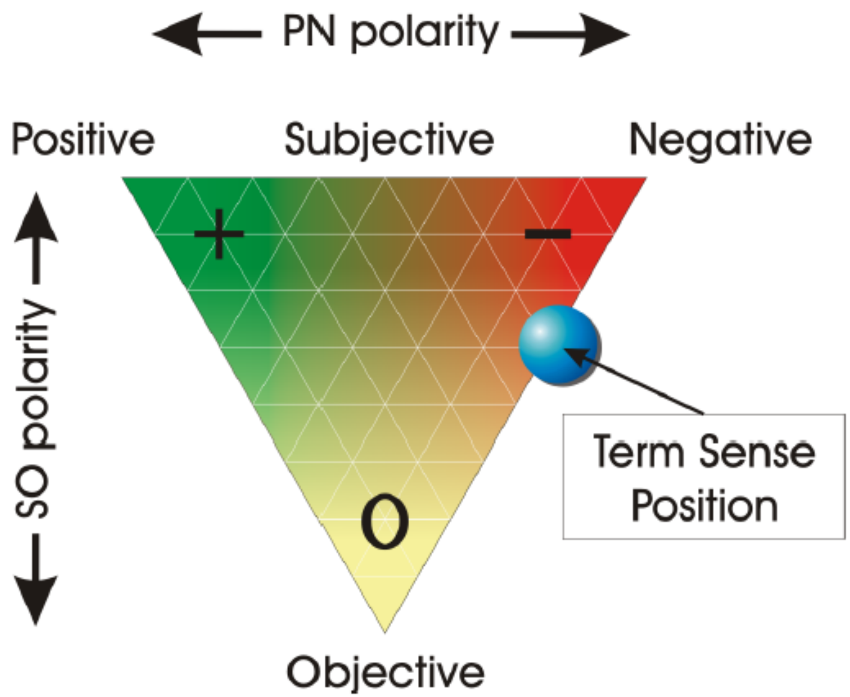
# LIWC: Linguistic Inquiry and Word Count

| 125 Affect | | | | | | 126 Posemo | | | 127 Negemo | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| abandon* | damn* | fume* | kindn* | privileg* | supporting | accept | freed* | partie* | abandon* | enrag* | maddening | snob* |
| abuse* | danger* | fuming | kiss* | prize* | supportive* | accepta* | freeing | party* | abuse* | envie* | madder | sob |
| abusi* | daring | fun | laidback | problem* | supports | accepted | freely | passion* | abusi* | envious | maddest | sobbed |
| accept | darlin* | funn* | lame* | profit* | suprem* | accepting | freeness | peace* | ache* | envy* | maniac* | sobbing |
| accepta* | daze* | furious* | laugh* | promis* | sure* | accepts | freer | perfect* | aching | evil* | masochis* | sobs |
| accepted | dear* | fury | lazie* | protest | surpris* | active* | frees* | play | advers* | excruciat* | melanchol* | solemn* |
| accepting | decay* | geek* | lazy | protested | suspicio* | admir* | friend* | played | afraid | exhaust* | mess | sorrow* |
| accepts | defeat* | genero* | liabilit* | protesting | sweet | ador* | fun | playful* | aggravat* | fail* | messy | sorry |
| ache* | defect* | gentle | liar* | proud* | sweetheart* | advantag* | funn* | playing | aggress* | fake | miser* | spite* |
| aching | defenc* | gentler | libert* | puk* | sweetie* | adventur* | genero* | plays | agitat* | fatal* | miss | stammer* |
| active* | defens* | gentlest | lied | punish* | sweetly | affection* | gentle | pleasant* | agoniz* | fatigu* | missed | stank |
| admir* | definite | gently | lies | radian* | sweetness* | agree | gentler | please* | agony | fault* | misses | startl* |
| ador* | definitely | giggl* | like | rage* | sweets | agreeab* | gentlest | pleasing | alarm* | fear | missing | steal* |
| advantag* | degrad* | giver* | likeab* | raging | talent* | agreed | gently | pleasur* | alone | feared | mistak* | stench* |
| adventur* | delectabl* | giving | liked | rancid* | tantrum* | agreeing | giggl* | popular* | anger* | fearful* | mock | stink* |
| advers* | delicate* | glad | likes | rape* | tears | agreement* | giver* | positiv* | angr* | fearing | mocked | strain* |
| affection* | delicious* | gladly | liking | raping | teas* | agrees | giving | prais* | anguish* | fears | mocker* | strange |
| afraid | deligh* | glamor* | livel* | rapist* | tehe | alright* | glad | precious* | annoy* | feroc* | mocking | stress* |
| aggravat* | depress* | glamour* | LMAO | readiness | temper | amaz* | gladly | prettie* | antagoni* | feud* | mocks | struggl* |
| aggress* | depriv* | gloom* | LOL | ready | tempers | amor* | glamor* | pretty | anxi* | fiery | molest* | stubborn* |
| agitat* | despair* | glori* | lone* | reassur* | tender* | amus* | glamour* | pride | apath* | fight* | mooch* | stunk |
| agoniz* | desperat* | glory | longing* | rebel* | tense* | aok | glori* | privileg* | appall* | fired | moodi* | stunned |
| agony | despis* | goddam* | lose | reek* | tensing | appreciat* | glory | prize* | apprehens* | flunk* | moody | stuns |
| agree | destroy* | good | loser* | regret* | tension* | assur* | good | profit* | argh* | foe* | moron* | stupid* |
| agreeab* | destruct* | goodness | loses | reject* | terribl* | attachment* | goodness | promis* | argu* | fool* | mourn* | stutter* |
| agreed | determina* | gorgeous* | losing | relax* | terrific* | attract* | gorgeous* | proud* | arrogan* | forbid* | murder* | submissive* |
| agreeing | determined | gossip* | loss* | relief | terrified | award* | grace | radian* | asham* | fought | nag* | suck |
| agreement* | devastat* | grace | lost | reliev* | terrifies | awesome | graced | readiness | assault* | frantic* | nast* | sucked |
| agrees | devil* | graced | lous* | reluctan* | terrify | beaut* | graceful* | ready | asshole* | freak* | needy | sucker* |
| alarm* | devot* | graceful* | love | remorse* | terrifying | beloved | graces | reassur* | attack* | fright* | neglect* | sucks |
| alone | difficult* | graces | loved | repress* | terror* | benefic* | graci* | relax* | aversi* | frustrat* | nerd* | sucky |
| alright* | digni* | graci* | lovely | resent* | thank | benefit | grand | relief | avoid* | fuck | nervous* | suffer |

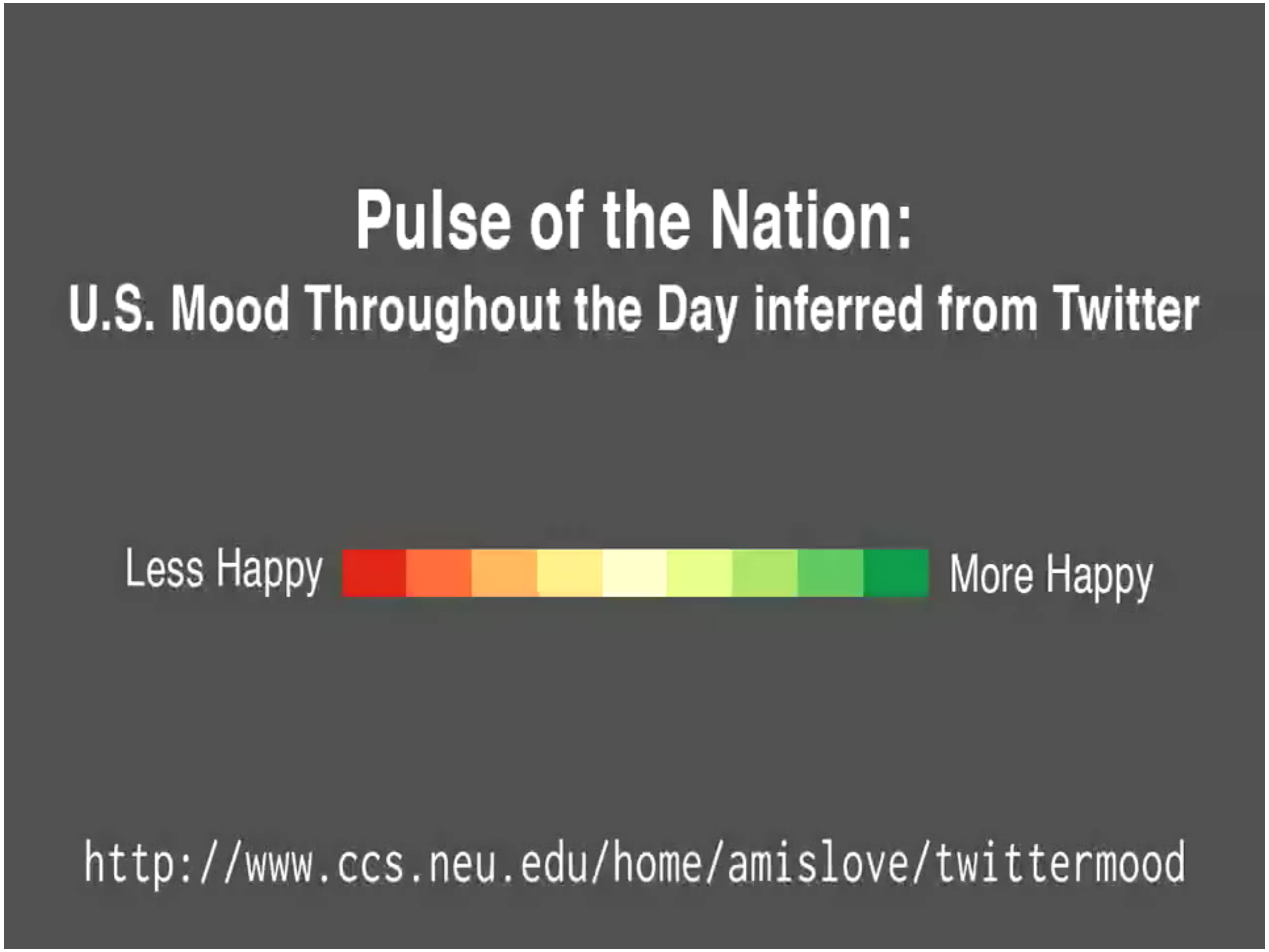
# ANEW: Affective norms for English words

| Description | Word No. | Valence Mean(SD) | Arousal Mean(SD) | Dominance Mean (SD) | Word Frequency |
|---|---|---|---|---|---|
| abduction | 621 | 2.76 (2.06) | 5.53 (2.43) | 3.49 (2.38) | 1 |
| abortion | 622 | 3.50 (2.30) | 5.39 (2.80) | 4.59 (2.54) | 6 |
| absurd | 623 | 4.26 (1.82) | 4.36 (2.20) | 4.73 (1.72) | 17 |
| abundance | 624 | 6.59 (2.01) | 5.51 (2.63) | 5.80 (2.16) | 13 |
| abuse | 1 | 1.80 (1.23) | 6.83 (2.70) | 3.69 (2.94) | 18 |
| acceptance | 625 | 7.98 (1.42) | 5.40 (2.70) | 6.64 (1.91) | 49 |
| accident | 2 | 2.05 (1.19) | 6.26 (2.87) | 3.76 (2.22) | 33 |
| ace | 626 | 6.88 (1.93) | 5.50 (2.66) | 6.39 (2.31) | 15 |
| ache | 627 | 2.46 (1.52) | 5.00 (2.45) | 3.54 (1.73) | 4 |
| achievement | 3 | 7.89 (1.38) | 5.53 (2.81) | 6.56 (2.35) | 65 |
| activate | 4 | 5.46 (0.98) | 4.86 (2.56) | 5.43 (1.84) | 2 |
| addict | 581 | 2.48 (2.08) | 5.66 (2.26) | 3.72 (2.54) | 1 |
| addicted | 628 | 2.51 (1.42) | 4.81 (2.46) | 3.46 (2.23) | 3 |
| admired | 5 | 7.74 (1.84) | 6.11 (2.36) | 7.53 (1.94) | 17 |
| adorable | 6 | 7.81 (1.24) | 5.12 (2.71) | 5.74 (2.48) | 3 |
| adult | 546 | 6.49 (1.50) | 4.76 (1.95) | 5.75 (2.21) | 25 |
| advantage | 629 | 6.95 (1.85) | 4.76 (2.18) | 6.36 (2.23) | 73 |
| adventure | 630 | 7.60 (1.50) | 6.98 (2.15) | 6.46 (1.67) | 14 |
| affection | 7 | 8.39 (0.86) | 6.21 (2.75) | 6.08 (2.22) | 18 |
| afraid | 8 | 2.00 (1.28) | 6.67 (2.54) | 3.98 (2.63) | 57 |

# ***sentiwordnet**.isti.cnr.it/*
# SentiWordNet

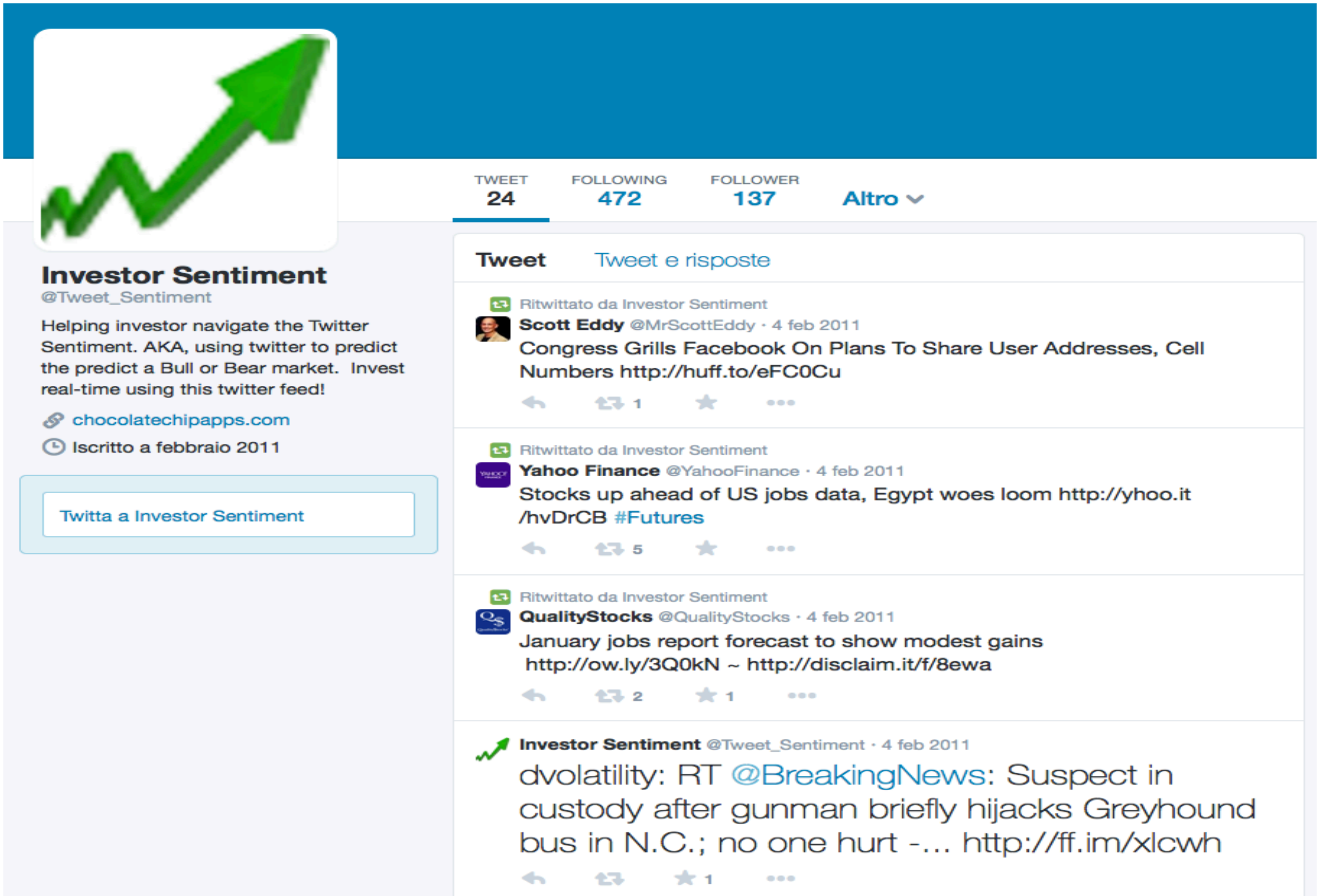


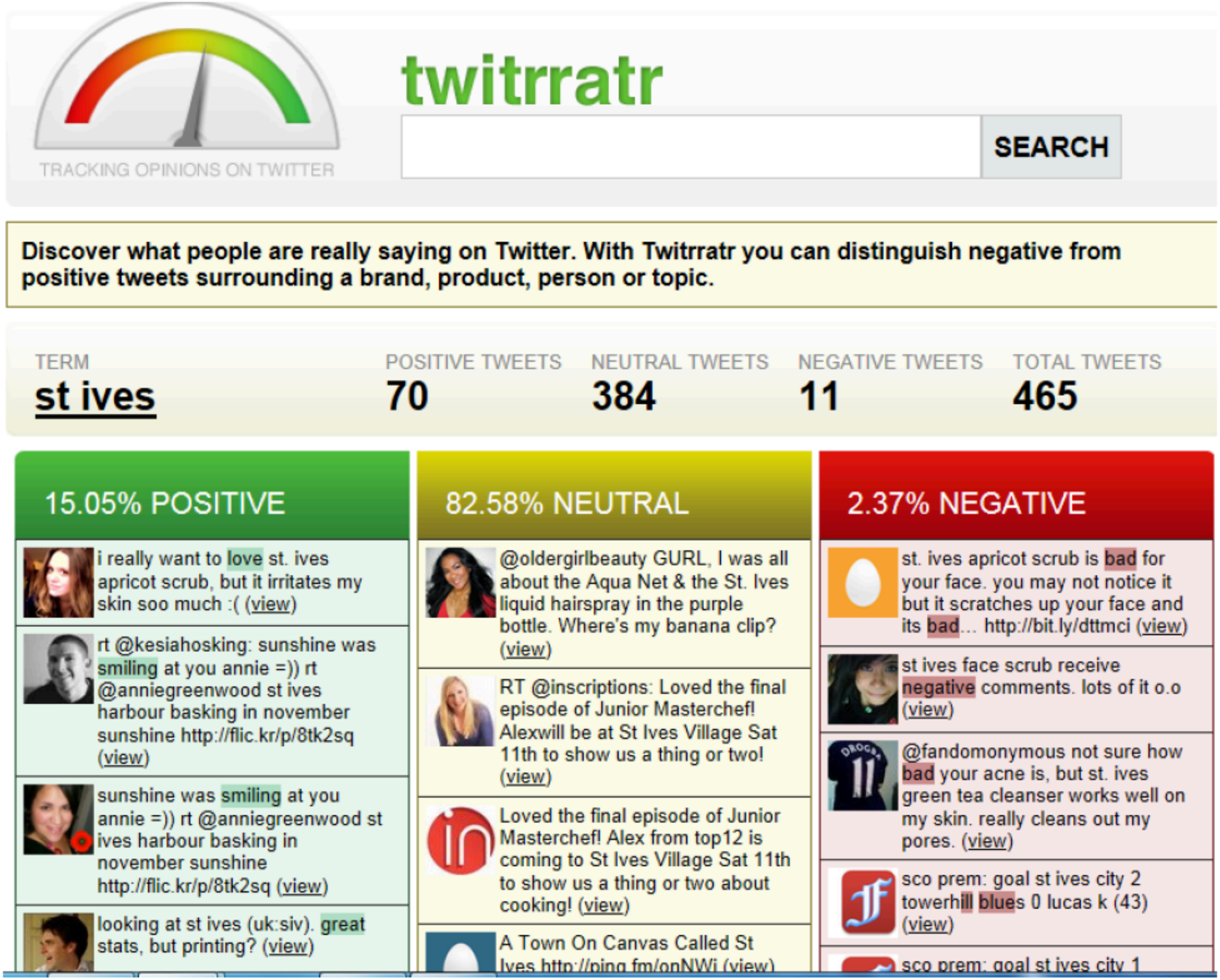
# Opinion-Mining Tools




http://www.ccs.neu.edu/home/amislove/twittermood/

# Twitter investor sentiment




TRACKING OPINIONS ON TWITTER
twitrratr
SEARCH
Discover what people are really saying on Twitter. With Twitrratr you can distinguish negative from positive tweets surrounding a brand, product, person or topic.
TERM
st ives
POSITIVE TWEETS
70
NEUTRAL TWEETS
384
NEGATIVE TWEETS
11
TOTAL TWEETS
465
15.05% POSITIVE
i really want to love st. ives apricot scrub, but it irritates my skin soo much :( (view)
rt @kesiahosking: sunshine was smiling at you annie =)) rt @anniegreenwood st ives harbour basking in november sunshine http://flic.kr/p/8tk2sq (view)
sunshine was smiling at you annie =)) rt @anniegreenwood st ives harbour basking in november sunshine http://flic.kr/p/8tk2sq (view)
looking at st ives (uk:siv). great stats, but printing? (view)
82.58% NEUTRAL
@oldergirlbeauty GURL, I was all about the Aqua Net & the St. Ives liquid hairspray in the purple bottle. Where's my banana clip? (view)
RT @inscriptions: Loved the final episode of Junior Masterchef! Alexwill be at St Ives Village Sat 11th to show us a thing or two! (view)
Loved the final episode of Junior Masterchef! Alex from top12 is coming to St Ives Village Sat 11th to show us a thing or two about cooking! (view)
A Town On Canvas Called St Ives http://ping.fm/onNWi (view)
2.37% NEGATIVE
st. ives apricot scrub is bad for your face. you may not notice it but it scratches up your face and its bad... http://bit.ly/dttmci (view)
st ives face scrub receive negative comments. lots of it o.o (view)
@fandomonymous not sure how bad your acne is, but st. ives green tea cleanser works well on my skin. really cleans out my pores. (view)
sco prem: goal st ives city 2 towerhill blues 0 lucas k (43) (view)
sco prem: goal st ives city 1

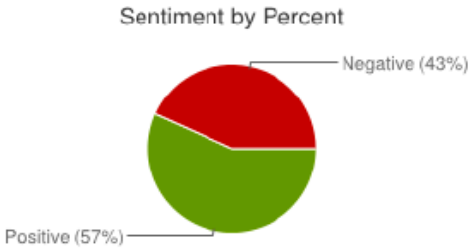

tweetfeel
Bonanza
Search
Try some Twitter trends: Romo Bonanza
4
3
= 57%
RT @BuildYourLoveUp: RT @BuildYourLoveUp: I wish @itsimreeeee went to the same school as me. I miss my best friend, and almost everyone at Bonanza sucks. ;/
shout out to the helicopter circling our school this morning with a spot light.. I love Bonanza http://t.co/j2EDX0cS
RT @ADReamGONe: RT @ADReamGONe: Man, I love Bonanza lmfao.
RT @ADReamGONe: RT @ADReamGONe: Man, I love Bonanza lmfao.

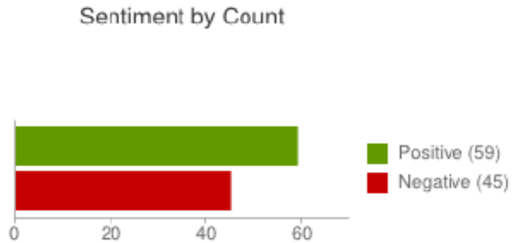
# Sentiment140

Tweet 353 | Like 140 | +1 74

microsoft | English | Search | Save this search

**Sentiment analysis for microsoft**




**Tweets about: microsoft**

Isaydumb: @Youporn, in my humble opinion you have nothing to do on the @Xbox Live. What the fuck is **@Microsoft** doing?!
Posted 46 seconds ago

Megan_Maracle: I hate this class. **#Microsoft** #die
Posted 2 minutes ago

dilwortha: @carasmith10 oh okay, you'll have to explain when i see you as i dont understand this disk haha. is it for **microsoft** project do you

Posted 5 minutes ago

jlebrech: @rsslldnphy it happens to be **microsoft** this time, but a superset is the next best thing from a compiled bytecode, as valid JS is also

http://www.sentiment140.com/

The results for this query are: Accurate

# Twitter Sentiment Visualization

# Opinion Finder

Multi-Perspective Question Answering

**Main** MPQA Home | **Corpora** News, debates, etc. | **Lexicons** Subj. clues, etc. | **Annotation** GATE, MPQA scheme | **OpinionFinder** Subjectivity detector

OpinionFinder

- Version 1.x
  - Version 1.5
  - Version 1.4
  - Sample Annotations
- Version 2.x

## OpinionFinder 1.x Release Page

### OpinionFinder 1.x Available versions

OpinionFinder 1.x relies on many external software packages (e.g. SUNDANCE, SCOL, BoosTexter) which are neither built nor supported by our group. Since OpinionFinder was originally released in 2005, there are some compatibility issues with versions of various software and packages. We have reports that these problems sometimes result in an exhausting and even unsuccessful installation process of OpinionFinder. Since many of the people involved in the original development have graduated and left the group, we do not currently have the resources to address these compatibility issues concerning the required external software packages. Although we do not have the resources to bring OpinionFinder 1.x fully up-to-date, we are currently working on a new version of OpinionFinder. OpinionFinder 2 is being written in Java and will be platform-independent.

LICENSE AGREEMENT

Version 1.5

- README - OpinionFinder 1.5
- Download OpinionFinder 1.5