ESSE: Exploring Mood on the Web

Sara Owsley Sood and Lucy Vasserman

Computer Science Department, Pomona College
185 East Sixth Street, Room 232
Claremont, CA 91711
sara@cs.pomona.edu, lhv02006@mymail.pomona.edu

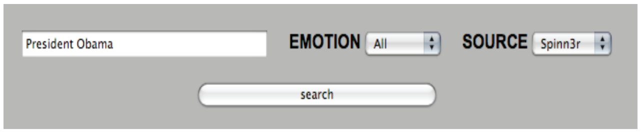
Abstract

Future machines will connect with users on an emotional level in addition to performing complex computations (Norman 2004). In this article, we present a system that adds an emotional dimension to an activity that Internet users engage in frequently, search. *ESSE*, which stands for *Emotional State Search Engine*, is a web search engine that goes beyond facilitating a user's exploration of the web by topic, as search engines such as *Google* or *Yahoo*! afford. Rather, it enables the user to browse their topically relevant search results by mood, providing the user with a unique perspective on the topic at hand. Consider a user wishing to read opinions about the new president of the United States. Typing "President Obama" into a *Google* search box will return (among other results), a few recent news stories about Obama, the Whitehouse's website, as well as a wikipedia article about him. Typing "President Obama" into a *Google Blog Search* box will bring the user a bit closer to their goal in that all of the results are indeed blogs (typically opinions) about Obama. However, where blog search engines fall short is in providing users with a way to navigate and digest the vastness of the blogosphere, the incredible number of results for the query "President Obama" (approximately 17,335,307 as of 2/24/09) (Google Blog Search 2009). *ESSE* provides another dimension by which users can take in the vastness of the web or the blogosphere. This article outlines the contributions of *ESSE* including a new approach to mood classification.

Introduction

Future machines will need to connect with users on an emotional level in addition to performing complex computations quickly (Norman 2004). For this reason *affective computing* (Picard 1997), building machines with *emotional intelligence* (Goleman 1997; Mayer 1993), is an important field of research within Artificial Intelligence today. An emotionally intelligent machine must be able to both identify emotions in its user and express emotions itself. In this article, we present a system that adds an emotional dimension to an activity that Internet users engage in frequently, search.

ESSE, which stands for *Emotional State Search Engine*, is a web search engine that goes beyond facilitating a user's exploration of the web by topic, as search engines such as *Google* or *Yahoo*! afford. Rather, it enables the user to browse their topically relevant search results by mood, providing the user with a unique perspective on the topic at hand. Consider a user wishing to read opinions about the new president of the United States. Typing "President Obama" into a *Google* search box will return (among other results), a few recent news stories about Obama, the Whitehouse's website, as well as a wikipedia article about him. Typing "President Obama" to a *Google Blog Search* box user a bit closer to their goal in that all of the results are indeed blogs (typically opinions) about Obama. However, where blog search engines fall short is in providing users with a way to navigate and digest the vastness of the blogosphere, the incredible number of results for the query "President Obama" (approximately 17,335,307 as of 2/24/09) (Google Blog Search 2009).



President Obama

EMOTION All

SOURCE Spinn3r

Figure 1: The *ESSE* interface - a textbox and a search button. More advanced features allow the user to select a source data set for the results and/or a focal mood of choice.

In appearances, *ESSE* resembles a typical search engine – a text box and a search button (see Figure 1). A user types in a topic of interest, for example "President Obama," and presses "search." Results are displayed to the user; the novel way in which the results are displayed and the user is able to navigate them is what distinguishes this system from the current state of the art. On the result page, the user sees three columns of results: angry blog posts, happy blog posts, and sad blog posts on the topic of "President Obama." The system does a first pass at digesting the blogosphere; that is, it filters the results down to the most emotional posts on this topic, highlighting the happiest, angriest, and saddest (see Figure 4). Provided that the user wants to navigate opinions on the topic, the system relieves the burden that the vastness of the blogosphere places on users. It brings us many steps closer to the goal of allowing users to feasibly take in opinions from differing schools of thought – differing emotional opinions on a topic.

Related Work

The blogosphere is filled with emotional experiences, stories and opinions. Much focus has been placed on large-scale processing of the blogosphere for marketing purposes (Glance 2005). The blogosphere provides a wealth of real stories, experiences and opinions about brands and products that are invaluable for companies. For this reason, companies like *Umbria*, *Cymphony*, *Neilsen BuzzMetrics*, and *Evolve24* (among many others) provide companies with a way to digest the blogosphere on a large scale, giving them graphical and other summarizations of trends in opinion in the blogosphere (Cymphony 2009; Evolve24 2009; Nielsen BuzzMetrics 2009; Umbria 2009). These systems are incredibly effective, and provide companies a way to better understand their customers through analysis of the prohibitively vast blogosphere. *MoodViews* is similar to the efforts made by these companies, but less focused on business-to-business applications, more focused on conveying mood trends in the blogosphere to users (Mishne 2006).

All of this work in marketing research and the blogosphere has spurred a flurry of work in sentiment analysis, work that has made great progress since its relatively recent start. Many systems have used movie and product reviews as training data to classify the sentiment (typically positive, negative or neutral) for a target document (Pang 2002; Sood and Owsley RTS 2007; Turney 2002). Innovation and increased accuracy of these systems has come from different machine learning approaches (Alm 2005) as well as new approaches to feature selection and dealing with differing emotional connotations across domains (Aue 2005; Owsley 2006). While researchers have explored other dimensions of sentiment (Bradley 1999; Mehrabian 1996), *ESSE* contributes to the space of sentiment analysis in that it provides a more detailed characterization of a target document (scores on dimensions of three moods based on Ekman's model of emotion while past systems provide a classification on the axis of positive/negative) with reasonable accuracy (Ekman 2003).

Other systems have made strides toward provide marketing researchers with meaningful emotional stories from the blogosphere, as opposed to summarizations of trends (Sood 2007). Towards a different goal, *Buzz* was a related system that extracted emotional stories from the blogosphere and included them in an autonomous theatrical performance engine (Sood 2008). As this is a growing research area, others have built systems with similar goals. Related work in this space includes the *Transient Life* system, which summarizes information about an individual's general state, but explicitly pulls this information though discussions with the individual (Smale 2006). Other systems like *We Feel Fine* and *Dumpster* unite people's emotional experiences by extracting passages from blogs in which individuals use phrases that typically indicate an expression of emotion such as "I feel." In both systems, the blog excerpts are presented in an artistic installation that users can explore online (Harris 2005; Levin 2005).

Towards a different goal, users have recently benefited from researchers' efforts to make the blogosphere more navigable and searchable. Work on search relevance and automated blog content tagging systems provide users with a way to get to more topically relevant posts (Sood and Owsley 2007). However, often the vastness of the topically relevant result space leaves most posts unread. *ESSE* seeks to provide a way for readers to digest the blogosphere, while giving them a different view of the results – bringing together and contributing to work in the space of sentiment analysis and search.

The System

While the bulk of our efforts and this article focus on one major component (the mood classifier), the system is actually comprised of three components:

1) The textual mood classifier that takes a document as input and returns set of scores reflecting the mood conveyed in the text.
2) A tfidf search engine backend, including an index of the spinn3r dataset – the 44 million blog posts made from August 1st to October 1st of 2008 (Burton et al. 2009).
3) The *ESSE* user interface.

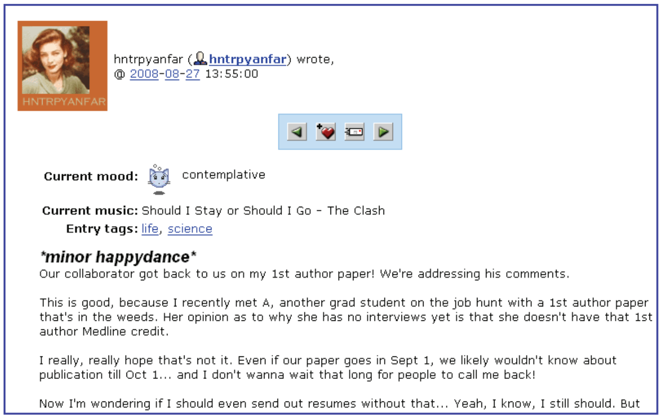
In the section that follows, the three components are described.

The Textual Mood Classifier

One of the core pieces of the *ESSE* system is a text-based mood classifier. The system is trained on a large set of blog posts labeled with the self-declared mood of the author. Using a vast set of features, this Naïve Bayes Classifier has achieved high accuracy and plays a central role in filtering the *ESSE* results. Aside from the diversity of features used in classification, this system is again unique in that it leverages clustering techniques in order to make use of a large training dataset. The section that follows describes how the mood classifier works; evaluation of this system is given in the "Evaluation" section.

Training Data

A *Naïve Bayes Classifier* requires a substantial set of labeled truth data for the system to learn from. Of the challenges in creating such a classifier, finding the appropriate dataset is of great importance. In this case, the training data is a corpus of more than six hundred thousand blog posts from the blog site *LiveJournal* (LiveJournal 2009). LiveJournal allows users to tag each post with a mood (e.g. – happy, anxious, angry, surprised); this label is an ideal truth-value for a mood classifier (see Figure 2).



hntrpyanfar (hntrpyanfar) wrote,
@ 2008-08-27 13:55:00

Current mood: contemplative

Current music: Should I Stay or Should I Go - The Clash

Entry tags: life, science

minor happydance*

Our collaborator got back to us on my 1st author paper! We're addressing his comments.

This is good, because I recently met A, another grad student on the job hunt with a 1st author paper that's in the weeds. Her opinion as to why she has no interviews yet is that she doesn't have that 1st author Medline credit.

I really, really hope that's not it. Even if our paper goes in Sept 1, we likely wouldn't know about publication till Oct 1... and I don't wanna wait that long for people to call me back!

Now I'm wondering if I should even send out resumes without that... Yeah, I know, I still should. But

Figure 2: LiveJournal post, tagged with a 'current mood' of 'contemplative.' (http://hntrpyanfar.livejournal.com/225723.html)

While the size of the LiveJournal blogs dataset is ideal, the site gives users one hundred and thirty moods to choose from, as well as allowing them to "write-in" their own mood. Using all one hundred and thirty plus moods in training and classification would hinder accuracy of the classifier and create feature frequency count databases too large for a system to use in real-time. Finally, and most importantly, it is not clear that there is a meaningful textual distinction between posts labeled as 'peeved' and 'annoyed,' for example. If such a distinction does not exist, then training a classifier to make such a distinction is not only mathematically impossible, but the effort would be futile and overfit to the examples in the dataset as some of the labels are used on very few posts. The smaller the number of labels, the more likely it is that such a distinction exists and can be learned by the system.

A first pass at compressing the set of labels involved a small user study, asking participants to place each of the one hundred and thirty mood labels into one of four buckets – happy, sad, angry or none of the above. However, after administering this survey to ten participants, it became clear that the responses were too dissimilar to be the basis of a statistical classification system. To preserve accuracy, the training dataset needed to be compressed into a smaller set of labels in a systematic way that would ensure statistical similarity within the training data with a particular label. The following section describes our efforts to reduce this space of mood labels using a K-Means Clustering approach.

K Means Clustering

To begin, we compressed the dataset by removing all posts that were labeled with "write-in" moods. Although this removed much of the noise in the data, the data required further compression. Specifically, we intended to build a system that could classify text as one of three moods (happy, sad and angry). To reach this goal, we sought to filter and compress this space of training data into 'happy' posts, 'angry' posts and 'sad' posts, discarding posts that fell outside of these three super categories.

While one could simply use the posts labeled as 'happy' for the 'happy' posts training data, we found that this set was relatively small and wasteful considering that posts labeled as 'ecstatic,' for example, should likely fall into the same category. To this end, we used *k-means clustering* to organize the one hundred and thirty moods labels into three groups: angry, happy, and sad and discard the mood labels that formed other clusters or were outliers.

To see why it was necessary to discard outliers, the following are "mood" labels from the *LiveJournal* dataset: 'hungry,' 'artistic,' 'sleepy,' 'blah,' 'working,' 'silly,' 'sore,' 'numb,' 'hot,' and 'okay.' As is shown in this list, several of the mood labels were meaningless for our purposes and would only introduce noise into the system. For this reason, outlying moods that do not fall neatly into the categories of 'happy,' 'sad,' or 'angry' were removed.

Given the original dataset of approximately six hundred thousand blog posts labeled with one of one hundred and thirty moods, the K-Means Clustering algorithm was used to determine which moods were similar enough to be grouped together and which were outliers. While one could intuitively hypothesize which labels should be clustered together, usage trends of each label might result in unexpected differences – for example, posts labeled as 'envious' and 'jealous' may be distinguishable – dissimilar enough to create noise in the data.

To carry out K-Means Clustering, each mood was represented as a vector of feature data; each position in the vector corresponded to a feature and the values were the total number of times that feature occurred in all posts tagged with that mood. The features used in our vectors were all unigrams that occurred across all blog posts in the entire dataset. K-Means Clustering was initiated on the 130 data points by using the data points for the 'happy,' 'sad,' and 'angry' labels as the initial 3 clusters.

The similarity calculation between data points (moods) was simply the distance between their corresponding vectors. The vectors were clustered into k groups based on their similarity. After running the algorithm multiple times, several distinct groups emerged. Of the one hundred and

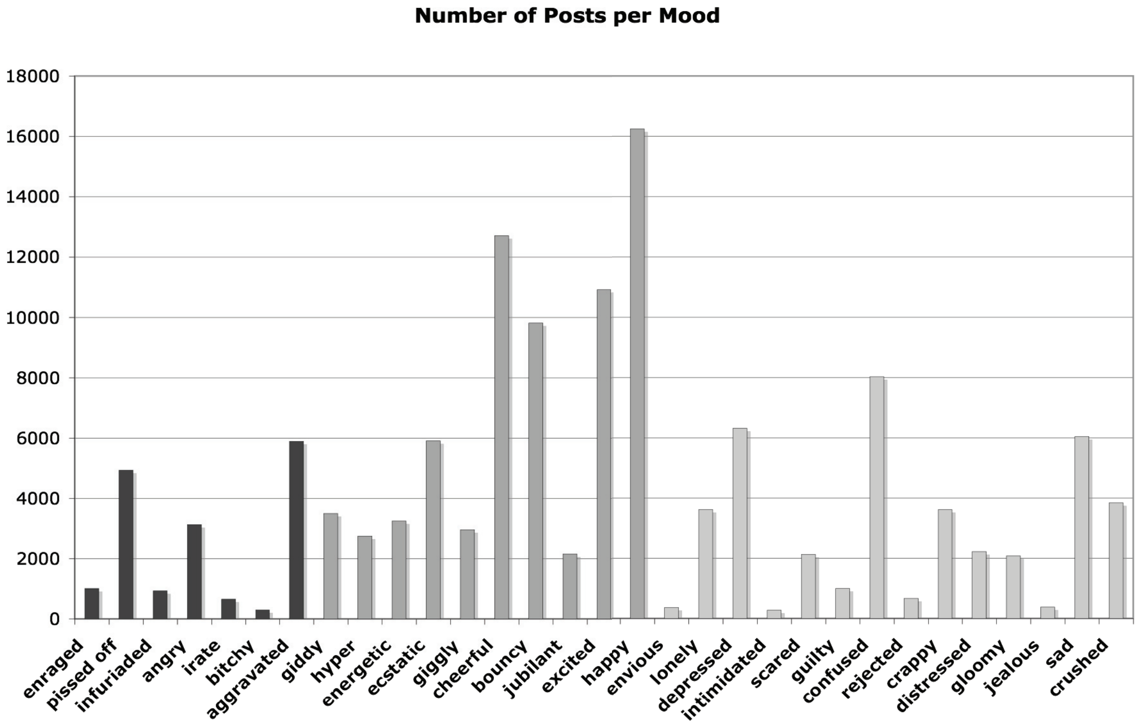


Figure 3: This graph shows the distribution of blog posts among mood labels. The horizontal axis gives the mood label, and the vertical axis gives the number of posts with that mood in the corpus. The bar color shows the groups of labels (angry, happy and sad).

thirty original mood labels, thirty-one of them fell clearly into three groups (which were representative of happy, sad, and angry moods). Only the posts tagged with those moods were used as training data. The final grouping of mood labels, ignoring outliers and small outlying clusters, is shown in Table 1 and the frequency of posts with those labels is shown in Figure 3. From the original six hundred thousand posts in the corpus, about one hundred and thirty thousand posts tagged with these thirty-one different moods, were used in the final training set.

Happy	Sad	Angry
Energetic	Confused	Aggravated
Bouncy	Crappy	Angry
Happy	Crushed	Bitchy
Hyper	Depressed	Enraged
Cheerful	Distressed	Infuriated
Ecstatic	Envious	Irate
Excited	Gloomy	Pissed off
Jubilant	Guilty	
Giddy	Intimidated	
Giggly	Jealous	
	Lonely	
	Rejected	
	Sad	
	Scared	

Table 1: The three prominent mood groups that emerged from K-Means Clustering on the set of LiveJournal mood labels.

Classifier

After compressing the dataset to a set of one hundred and thirty thousand blog posts that fell neatly into the categories of happy, sad and angry, the classifier itself was trained using this data. The system uses a Naïve Bayes approach in order to calculate the conditional probability of a document *d* being a member of a class *c*, where the three possible classes are *happy*, *sad* and *angry*. Given these three probabilities, a document is then classified as the class with the highest conditional probability.

By Bayes theorem, the conditional probability of a class *c* given a document *d* is calculated as the prior probability of the class *c* multiplied by the probability of each feature in *d* given that class *c*. The conditional probability of a feature *f* occurring given that a document is of class *c* is equal to the training frequency of feature *f* in class *c* divided by the sum of all of frequencies of features in class *c*. The prior probability of a class *c* is simply equal to the fraction of training documents from class *c*.

The probability that a target document *d* is "happy" given the set of *n* features of *d,* which we'll call *f* is:

$$P(happy\,|\,f) = P(happy) * \prod_{i=1}^{n} P(f_i\,|\,happy)$$

The first term on the right side of the equation is the prior probability of any document being "happy." The second term on the right side of the equation is the product of the conditional probabilities of each of the features occurring given that a document is "happy".

Finally, after the three probabilities are calculated for a given document, the system must determine the classification. The highest probability class is interpreted as the most likely class and returned as the classification of the document.

Features

Along with the data used in training, the features selected for training and classification are the most important part of a text classification system. The system was originally trained using unigrams as the sole features, but was quickly expanded to include many other classes of features that increased system accuracy.

In addition to standard classification features such as unigrams, bigrams and stems, the system includes features that are specific to the nature of blogs. A corpus of emoticons, emotional faces created from sequences of punctuation (e.g. :-) and :/), was used as a feature set. People often use emoticons to explicitly state their mood, so it is clear that emoticons are a powerful feature for mood classification. One drawback is that the coverage of this feature set in the blogosphere is small, however, when present, emoticons are quite indicative of the authors mood.

A list of "Internet slang," such as "lol" (which means 'laughing out loud') and "omg" (which means 'oh my god'), was gathered and used as a feature set in the classification system. While these terms would already have been used as individual features in the unigrams feature set, keeping them as a separate feature set enables more weight to be placed on them in the probability calculation. Similarly, a list of known highly emotional terms was gathered and used as a separate feature set. Again, while emotional terms are counted in the unigram feature set, giving them distinction as a separate feature set allows them to carry more weight in the classification.

These last three feature sets (in Table 2) are typically only present in a small portion of texts. However, emotion words, Internet slang, and emoticons are generally highly indicative of the mood of the text and therefore are worthy to be included as features in the final system. The classifier is most accurate using a combination of all six features. The classification accuracy with different feature set combinations is given in the "Evaluation" section. Given that classification accuracy differs by feature sets in differing document types, it was critical that the features and weights used in classification were left configurable. The final version of the classifier allows the user to set how important each feature will be in classification on a scale of zero to ten. This will allow for customization based on the type of document being classified (emoticons and internet slang will be much more useful for classifying blogs or online conversations than for news articles).

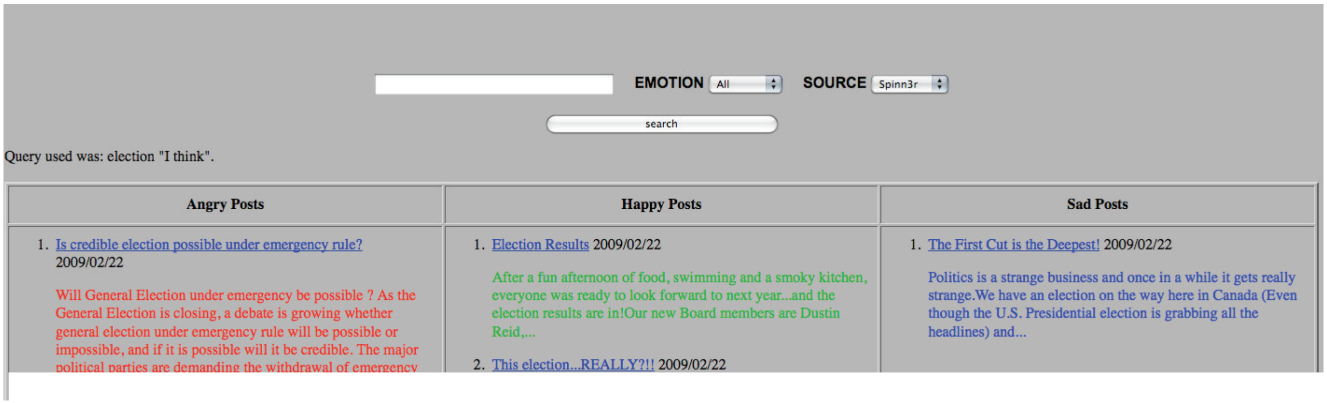
Feature Set	Description
Unigrams	Single terms including interior punctuation (typically apostrophes and dashes) and excluding stop words
Bigrams	Two word phrases, again including interior punctuation and excluding bigrams that include stop words
Stems	Using a Python based implementation of Porter's stemmer, words are replaced with and represented in the system by their stem
Emotion Words	Words that describe or identify emotions
Emoticons	Emotional faces made from punctuation such as '☺'
Slang	Internet slang such as 'lol' or 'omg'

Table 2: The collection of feature sets used in the mood classification system.

The Spinn3r Dataset and Search Backend

Given an accurate mood classification system, one might imagine it to be simple to configure the classifier as a search filter, thus creating a mood-based retrieval system. However, the challenge lies in the fact that in order to classify the mood for a potential result, the entire content of that page must be downloaded and analyzed. Much like a typical web-based retrieval system, to avoid this cost, pages could be crawled and their mood indexed along with the representation stored for search indexing.

Alternatively, the presence of a massive dataset from www.spinn3r.com enabled the *ESSE* system to be built, performing mood classification and result filtering on the fly (Burton et al. 2009). Because the dataset (including textual content), search system, and mood classification system all exist on the same server, the filtering retrieval system was made possible. The dataset not only allows access to the content of a blog post (beyond the summary and title typically made available through search APIs) but the closed nature of the dataset allows for experimentation while still being vast enough to provide breadth and depth



EMOTION All SOURCE Spinn3r

Query used was: election "I think".

Angry Posts	Happy Posts	Sad Posts
1. Is credible election possible under emergency rule? 2009/02/22 Will General Election under emergency be possible ? As the General Election is closing, a debate is growing whether general election under emergency rule will be possible or impossible, and if it is possible will it be credible. The major political parties are demanding the withdrawal of emergency	1. Election Results 2009/02/22 After a fun afternoon of food, swimming and a smoky kitchen, everyone was ready to look forward to next year...and the election results are in!Our new Board members are Dustin Reid,... 2. This election...REALLY?!! 2009/02/22	1. The First Cut is the Deepest! 2009/02/22 Politics is a strange business and once in a while it gets really strange.We have an election on the way here in Canada (Even though the U.S. Presidential election is grabbing all the headlines) and...

Figure 4: A set of ESSE results for the query "election."

of topical coverage.

The Spinn3r dataset contains 44 million blog posts made from August 1st to October 1st of 2008. The posts are organized in XML files, with one to three XML files per date in the range mentioned above. The *ESSE* system includes a module to preprocess the Spinn3r dataset; beginning with the XML representation of the posts provided by Spinn3r, and ending with serialized Python based representation of the raw data from the posts including the url, title, summary and text (cleaned up and with html markup removed). This data was used as the base of the search engine, and ultimately the *ESSE* interface.

After preprocessing the Spinn3r data, the posts must be indexed by the TFIDF search engine. The search engine, implemented in Python, uses the standard TFIDF document representation approach, giving weight to words that occur frequently in a document, but devaluing the importance of words that occur frequently across a corpus (Salton 1983; Salton and Buckley1988). Indexing a corpus involves creating a "term frequency" table that keeps track of the frequency that a word such as "cat" appears in each document in the corpus. The "document frequency" table stores the number of documents that each word in the corpus appears in. The indexing stage stores tables of these values to enable a real-time search engine. After indexing, the final system is able to take a query and find the top matching blog posts by looking of the term frequency and document frequency table entries for the query terms.

The *ESSE* Interface

The goal of the *ESSE* system is to enable users to take in the vastness of blog entries on a single topic by breaking down the set of relevant blogs into smaller sets. In particular, the *ESSE* system uses mood conveyed in a blog post as the metric for dividing the result set. This measure not only provides a clear distinction between blog posts but also raises the user's awareness of differing opinions on the topic at hand.

Towards these goals, the interface was to be simple. We cannot aide a user, simplifying the complex task of navigating the blogosphere, without a simple interface. The interface must also meet users expectations of a "search engine" – a text box and a search button, that, when pressed will present the user with a set of results in textual and hyperlinked format. Because the system back-end was Python based, Django was used to create a front-end system.

The *ESSE* system can operate in two modes. Both modes begin at the main page shown in Figure 1. The first is most similar to a standard search engine. The user types in a term(s) of interest and presses "search." She/he is presented with a set of results from the blogosphere, divided into three groups, happy posts, angry posts and sad posts (see Figure 4). To emphasize this division, the text of each post summary conveys the mood that is characteristic of that post as a whole (angry posts are red, sad posts are blue and happy posts are green). In the second mode, users not only specify their term(s) of interest, but also select a mood from a dropdown menu. This selection specifies the mood of the result set they will see; they will not see results characteristic of the other two moods.

Evaluation

At the core of the *ESSE* system is the text based mood classification system. In addition to the design of the interface that enables the user to navigate the result space, the effectiveness of *ESSE* as an interface for digesting the vastness of the blogosphere depends greatly on the accuracy of the mood classification system itself. The system evaluation that follows focuses on just how effective the mood classification system is in the task of extracting the author's mood from the text of a blog post.

To evaluate the accuracy of the mood classification system, four test classifiers were trained on different combinations of feature sets. Each classifier was trained on 95% of the data, and tested on the remaining 5% with separate tests performed on the testing data for each of the three major mood groups (angry, happy and sad). Precision, recall and f-measure scores were calculated for each classifier, as seen in Table 3. The average f-measure was taken as an overall score for each classifier. Given a selection of text, each classifier returns a classification of either 'happy,' 'sad,' or 'angry.' Given the three possible classifications, the baseline performance would be 0.33.

In analyzing the results in Table 3, you will notice that only one of the test systems involved the three innovative feature types (Internet slang, emoticons and emotional words). The decision not to build a classifier solely based on each of these feature sets was due to the fact that they provided little coverage in the blogosphere. That is, most of the blog posts do not include these features. However, as you can see from the performance of the mood classifier that uses all six features, it is clear that including these feature sets is beneficial as when these features are present, they serve as good indicators of the mood of the blog post. The final system, shown at the bottom of Table 3 uses an ensemble of classifiers, each weighted by their importance, in order to reach a final classification. They weights are:
Unigrams – 9, Bigrams – 10, Stems- 9, emotion words – 1, Internet slang – 1, and Emoticons -1. This configuration of weights is the default for the system and performs with an average f-measure of 0.661, but the standalone mood

classification system allows users to alter these weights as desired.

	recall	precision	f-measure
Mood classifier using unigrams only.			
angry tests	0.499	0.465	0.481
happy tests	0.751	0.813	0.781
sad test	0.654	0.596	0.624
average	0.635	0.625	**0.630**
Mood classifier using bigrams only.			
angry tests	0.503	0.491	0.497
happy tests	0.763	0.821	0.791
sad test	0.67	0.602	0.634
average	0.645	0.638	**0.642**
Mood classifier using stems only.			
angry tests	0.502	0.457	0.478
happy tests	0.747	0.808	0.776
sad test	0.646	0.594	0.619
Average	0.632	0.620	**0.626**
Mood classifier using a weighted sum of all six different feature sets.			
angry tests	0.577	0.491	0.531
happy tests	0.751	0.84	0.793
sad test	0.685	0.62	0.651
Average	0.671	0.650	**0.661**

Table 3: Recall, Precision and F-Measure values for four test versions of the mood classification system.

You will also notice that the f-measure varies significantly by mood, with happy being much easier to classify than sad or angry. This is likely caused by the fact that there was more training data for happy texts. In addition, happy is a positive mood while sad and angry are both negative, and so identifying happiness is a much easier task than distinguishing sad and angry, even for humans.

Further future evaluation will focus on the effectiveness of the *ESSE* interface itself including a user study to drive improvements in the design and functionality.

Future Work

Future work will involve expansions and improvements of the capabilities of both the mood classifier as well as the *ESSE* interface.

The mood classifier currently classifies documents as 'happy,' 'sad,' or 'angry.' While this is more detailed than sentiment analysis, we wish to further expand the system to data from a larger set of moods, but without loss of classification accuracy. In a similar effort, we intend to add a dimension of 'valence' to the classifier so that the system can accurately classify documents as 'neutral' when appropriate. The system currently uses a vast collection of features where feature weights have default values but are also configurable. We plan to enhance the system to automatically detect which feature sets are most appropriate for a target document, based on the presence of particular features (Internet slang, emoticons, etc). Finally, we plan to perform a more detailed evaluation that analyzes system errors in order to improve system accuracy.

As mentioned in the section above, we intend to perform an evaluation of the effectiveness of the *ESSE* interface itself. The results of such a study will drive further innovations in functionality and design. In particular, we plan to prototype and implement an interface that will allow users to not only get search results divided by mood (as the current interface allows), but to move about the mood axes in real-time, allowing exploration of a larger result space driven by the user.

Acknowledgements

The authors would like to acknowledge and thank the Norris Foundation for their continued support of this work and the Artificial Intelligence Laboratory at Pomona College. Additionally, the authors are grateful to Spinn3r for their service to the weblog research community through the publishing of this dataset.

References

Alm, C.O., Roth, D., and Sproat, R. *Emotions from text: machine learning for text-based emotion prediction*. In Proceedings of HLT/EMNLP, 2005.

Aue, A. and Gamon, M. *Customizing sentiment classifiers to new domains: a case study*. RANLP, 2005.

M. M. Bradley and P. J. Lang. *Affective norms for English words (ANEW): Stimuli, instruction manual, and affective ratings*. Technical Report C-1, Center for Research in Psychophysiology, University of Florida, Gainesville, Florida, 1999.

K. Burton, A. Java, and I. Soboroff. *The ICWSM 2009 Spinn3r Dataset*. In Proceedings of the Third Annual International Conference on Weblogs and Social Media (ICWSM 2009), San Jose, CA, May 2009.

Cymfony. http://www.cymfony.com/, 2009.

Ekman, P. *Emotions Revealed: Recognizing Faces and Feelings to Improve Communication and Emotional Life*. Henry Holt and Company, New York, NY: 2003.

Evolve24, http://evolve24.com/, 2009.

Glance, N., Hurst, M., Nigam, K., Siegler, M., Stockton, R. and Tomokiyo, T. *Analyzing online discussion for marketing intelligence*. In Proceedings of the 14th International Conference on the World Wide Web, pages 1172 to1173, Chiba, Japan, 2005.

Goleman, D. *Emotional Intelligence: Why It Can Matter More Than IQ*. Bantam, New York: 1997.

Google Blog Search, http://blogsearch.google.com/, 2009.

Harris, J. and Kamvar, S. *We feel fine*. http://www.wefeelfine.org/, 2005.

Levin, G., Nigam, K., and Feinberg, J. *The dumpster*. http://artport.whitney.org/commissions/thedumpster/, 2005.

Liu, H., Lieberman, H., and Selker, T. *A model of textual affect sensing using real-world knowledge*. In Proceedings of the 8th international conference on Intelligent user interfaces, 2003.

LiveJournal. http://www.livejournal.com/, 2009.

Mayer, J.D. & Salovey, P. *The intelligence of emotional intelligence*. Intelligence, 17, 433-442, 1993.

Mehrabian, A. *Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in Temperament*. Current Psychology, Vol. 14, No. 4. (21 December 1996), pp. 261-292.

Mishne, G. and de Rijke, M. *Moodviews: Tools for blog mood analysis*. In AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs, 2006.

Nasukawa, T. and Yi, J. *Sentiment analysis: Capturing favorability using natural language processing*. In Proceedings of the 2nd International Conference on Knowledge Capture, 2003.

NielsenBuzzMetrics http://www.nielsenbuzzmetrics.com/, 2009.

Norman, D. *Emotional Design: Why we love (or hate) everyday things*. Basic Books, New York: 2004.

Owsley, Sara, Sood, Sanjay, Hammond, K. *Domain Specific Affective Classification of Documents*. AAAI Spring Symposia Computational Approaches to Analyzing Weblogs, 2006.

Pang, B., Lee, L. and Vaithyanathan, S. *Thumbs up? sentiment classification using machine learning techniques*. In Proceedings of EMNLP, pages 79 to 86, 2002.

Picard, R.W. Affective Computing. MIT Press, Cambridge, 1997.

Salton, G. *Introduction to Modern Information Retrieval*, McGraw-Hill, 1983.

Salton, G. and C. Buckley *Term-weighting approaches in automatic text retrieval*. Information Processing and Management. 24(5): 513 to 523, 1988.

Smale, S. and Greenberg, S. *Transient life: collecting and sharing personal information*. In OZCHI ’06: Proceedings of the 20th conference of the computer-human interaction special interest group(CHISIG) of Australia on Computer-human interaction: design: activities, artefacts and environments, pages 31–38, New York, NY, 2006.

Sood, Sanjay, Owsley, Sara, Hammond, K. and Birnbaum, L. *Reasoning Through Search: A Novel Approach to Sentiment Classification*. Northwestern University Tech Report Number NWU-EECS-07-05, 2007.

Sood, Sanjay, Owsley, Sara, Hammond, K., and Birnbaum, L. *Tag Assist: Automatic Tag Suggestion for Blog Posts*. International Conference on Weblogs and Social Media, 2007.

Sood, Sara Owsley. *Compelling Computation: Strategies for Mining the Interesting*. PhD Thesis, 2007.

Sood, Sara Owsley. *Buzz: Mining and Presenting Interesting Stories*. The International Journal of Art and Technology V1 N1, 2008.

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

Umbria. http://www.umbrialistens.com/, 2009.