**Literature Survey:**

1. **Opinion Mining and Sentiment Analysis**

Bo Pang and Lillian Lee (2008)

* A major part of gathering information is to find out what people think and what their preferences are.
* Studies show that approximately 81% of individuals show interest in the reviews about products online. Thus the product reviews affect/influence their opinions to a great extent.
* The vendors of online products are now paying more attention towards the analysis of the reviews so that they can improve their service in this cut-throat competition.
* This has caused a sudden interest towards the field of machine learning for the computational treatment of opinion.
* New ways to achieve this goal are discovered everyday by means of technology and it is slowly enabling the computers to recognize and express some emotions.
* There are a lot of huge datasets that can be used for the development of the product.
* This can also have bad impacts on the company as their rivals can fill in false reviews to tamper with the original data thus creating a lot of confusion and ultimately causing loss for the company.

1. **Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis**

Theresa Wilson, Janyce Wiebe and Paul Hoffmann(2005)

* This paper gives us a method to find the contextual polarity of any phrase.
* First all the words are assigned a value or weight which determines whether the word is positive negative or neutral.
* The next step is to take the entire text and find out an aggregate polarity of the text.
* Finding the contextual polarity of a text can be a very tedious task as the polarity of a text can be influenced by modality.
* “The movie was **not only** good but amazing”. Sentences like this have a positive polarity, but the computer might not be able to correctly identify it as there is a negative word in it.
* This problem can be corrected to an extent by assigning different values to the words based on their impact or intensity.
* This way words such as “Good”, “Nice” may get lower value as compared to “Awesome”, “Magnificent”.
* The results from this research paper indicate that a simple classifier can give accuracy of up to 50%.

1. **Social Network User’s Content Personalization based on Emotions**

Anthoniraj Amalnathan and Margret Anouncia(2015)

* The content/review any user posts online mostly can be used to predict his/her behaviour.
* This paper uses twitter as the dataset. We extract a large dataset of tweets from twitter.
* The tweets are cleaned of all the useless characters like @, #, URL.
* Next step is to remove the Stop Words or the irrelevant words from the text. Once done we are left with text which is ready for training.
* Now the text is analysed and the words are segregated according to their category (noun, adverb, verb etc.)
* Next step is to test the tweets and to classify them according to the emotions the user has expressed in them. A decision tree is constructed using the words.
* This then allows the predictor to classify the tweet based on the type of word or expression contained.
* This algorithm shows an accuracy of 83% when tested of a dataset of 2000 tweets.
* This gives us an insight on the user’s behaviour thus allowing us to make changes to our product based on his preferences.

1. **Sentiment Analysis: Capturing Favorability Using Natural Language Processing**

Tetsuya Nasukawa and Jeonghee Yi (2003)

* This paper aims at extracting the sentiment from a given text with different values of polarities instead of just classifying the entire text as positive or negative.
* This method allow us to find the local sentiments, or sentiments from different part of the text. Thus if any review has both good and bad things it will not just classify it as one but rather provide us with both options.
* We cannot find out the correct sentiment of a text until we have the polarity and the strength of each good/bad word.
* The first step was to divide the text into fragments. The fragments contain main text to be analysed and a window of some words on either side to have a context.
* The minimum words required around the main text were found out using preliminary tests.
* POS tagging used to disambiguate the text from words that can have two meanings.
* Once the POS is assigned to the text, a shallow parser is used to find the phrase boundaries. Then the subjects and the objects are binded to predicates.
* After the text is passed through the shallow parser the subjects are assigned a sentiment term.
* Then the sentiment polarity is assigned to the subjects.
* The results show about 95% precision and as the datasets are increased the precision can go down to 75%. Which is still very good.

1. **Sentiment Analysis of Blogs by Combining Lexical Knowledge with Text Classification**

Prem Melville, Wojciech Gryc and Richard D. Lawrence (2009)

* The large amount of data on the web has made way for the companies that are interested in monitoring what people are saying about specific products and services.
* The base line in this algorithm is the Lexical Classification. We just find out the occurrences of good and bad words add up their polarities and then classify the text as good or bad.
* The second approach is to use Feature Supervision to assign a lexicon to each class.
* Then we just have to compare the similarity between the text and the class and the class with the highest similarity is trained using a Bayes Classifier.
* We can use pooling distributions to combine the data from two different sources.
* Since in this case we have absence of background data thus we try to estimate the class priors from the training data.
* The dataset used in this implementation was from IBM for the list of words and the text was extracted from different sources online.
* The results show better results for pooling distributions.
* In the results the datasets perform better when combined together than when used in isolation.

1. **Distributed Representations of Words and Phrases and their Compositionality**

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean (2013)

* In this paper, the authors introduce novel techniques to improve the training speed and the quality of the word vectors.
* Also combined collocations into a single token to be fed into the skip-gram model.
* Subsampling of the frequent words to learn more regular word representations.
* Techniques introduced for this are hierarchical softmax and negative sampling.
* To train the skip-gram model, a large dataset consisting of various news articles (an internal Google dataset with one billion words) where all words that occurred less than 5 times in the training data were discarded which resulted in an a vocabulary size of 692K.
* Got an accuracy of 61% on using negative samples with 15 negative samples. Rest of the results can be seen in the paper.
* An interesting result from their research work is that the word vectors can be meaningfully combined using simple vector additions.

1. **Twitter Sentiment Analysis with Deep Convolutional Neural Networks**

Aliaksei Severyn, Alessandro Moschitti (2015)

* The authors introduced a new model for initializing the parameter weights of the convolutional neural network to avoid injecting additional features.
* Initially used an unsupervised neural language model to train initial word embeddings that are further trained tuned by a deep learning model
* The model is trained using the supervised training data recently made available by the official system evaluation campaign on Twitter Sentiment Analysis organized by Semeval 2015.
* The model could be ranked among the first two positions in both the phrase level subtask and among the message level subtask B.
* Used a sentence matrix using pre-trained word embeddings and which was fed as input to the convolutional neural network.
* The network initialization process includes the use of distant supervised data to further refine the weights of the network passed from the completely unsupervised neural language model.
* In the future, the authors plan to apply deep learning based models to other IR applications.

1. **Twitter Sentiment Analysis: The Good the Bad and the OMG!**

Efthymios Kouloumpis, Theresa Wilson, Johanna Moore (2011)

* The authors investigated on how linguistic features could be used to analyze the sentiment of Twitter messages.
* Quickly identifying relevant training data that could be useful to understand the context of the topic was the highlight of their paper for which they used Twitter hashtags in order to identify positive, negative and neutral tweets which could be used to train the three way sentiment classifiers.
* For development and training, they used the hashtagged data set (HASH) and the emoticon data set (EMOT)
* For evaluation they used a manually annotated data set produced by the iSieve Corporation (ISIEVE)
* Selected only those hash tags from the HASH dataset whose occurence frequency was greater than 1000. From them the top hash tags were shortlisted which were later classified into positive, negative and neutral tweets.
* Preprocessing of the dataset primarily included tokenization, normalization and POS tagging.
* Features used by the authors for classification were n-gram features, lexicon features, part-of-speech features and micro-blogging features.
* 10% data was set aside for validation (paramater tuning and n-gram feature selection) and rest of the data was fed into an AdaBoost\_MH classifier with 500 rounds of boosting.
* They found out that the best performance on the test set was found when the features used were from n-gram, lexicon and microblogging
* Also they found out that POS features reduced the performance of the model which needed later investigation.

1. **Twitter Sentiment Classification using Distant Supervision**

Alec Go, Richa Bhayani, Lei Huang (2009)

* The authors propose automatic classification of the sentiment of Twitter messages into positive or negative classes with respect to the query term.
* They used a method called distant supervision wherein the training data consists of tweets with emoticons that serve as noisy labels
* They use a litmus test to judge whether a given tweet has a neutral sentiment or not that is if the tweet could ever appear as a frontpage newspaper headline or as a sentence in Wikipedia, tag that tweet as one with neutral sentiment. The authors did not use these tweets for classification purposes.
* The machine learning classifier used for this task were Naive Bayes, Maximum Entropy and Support Vector Machines.
* The features used were unigrams, bigrams, unigrams and bigrams , and unigrams with part of speech tags.
* Various preprocessing steps were applied and 800,000 positive and negative tweets were collected.
* Test set was manually collected which comprised of 177 negative tweets and 182 positive tweets.
* Got a maximum accuracy of 82.7% using Naive Bayes classifier
* POS tagged feature extractors were not found to be useful.
* Future work presented by them were understanding the semantics of the messages, domain specific tweets, handling neural tweets, internationalization (use of other languages apart from English) and utilizing emoticon data in the test set.

1. **Exploiting Emoticons in Sentiment Analysis**

Alexander Hogenboom, Daniella Bal, Flavius Frasincar, Malissa Bal, Franciska de Jong, Uzay Kaymak (2013)

* The authors presented how we can exploit ‘emoticons conveying sentiment’ by manually creating emoticon sentiment lexicon in order to beat state-of-the-art sentiment classification methods.
* Evaluation was conducted on 2080 Dutch tweets and forum messages, which all contained emoticons and were manually annotated for sentimetn analysis.
* Perform sentiment analysis by classifying it as either positive or negative
* The authors propose a pipeline to perform sentiment classification which included segmentation, emoticon detection, preprocessing, word typing and merger of emoticon based sentiment analysis and text based sentiment analysis into sentiment aggregation to rate the documents.
* For text based sentiment analysis, a semantic score is assigned based on average of the scores assigned to the various emoticons as derived from the emoticon sentiment lexicon.
* After determining the sentiment conveyed by each individual text segment, all the text segments are recombined into a single document which can comprise of segments; with and without emoticons.
* The document sentiment score is then calculated as a weighted average of all segment level sentiment scores. Results can be found in the paper.
* Limitations of their approach include polarity classification errors due to interpretation of human readers and their preference for certain aspects of a text over others.
* Future work includes to analyze positioning of the emoticons in the text.