Literature review: papers

**Paper 2 – Sentiment Analysis in Social Media and Its Application: Systematic**

**Literature Review**

* Main Themes
  + literature review
  + methods, platforms used, and applications
* Brief Content Summary
  + Most articles apply opinion lexicon
  + Mainly twitter
  + Sentiment analysis in world events, healthcare, politics, and business
  + Uses natural language processing
  + Sentiment analysis divided into three levels: sentence level, document level, feature level.
  + Which means classifying opinion either from sentence, document, or features into positive or negative sentiment
  + 2 main approaches, machine learning and lexicon based
  + Machine learning uses algorithms, lexicon based counts the number of positive or negative words
  + Lexicon based things aren’t great because they depend on how good the dictionary is and has no nuance like sarcasm and negation
  + The positives of it are that it is simple and easy to transfer to different languages
  + Machine learning
    - Most used method: SVM and Naïve Bayes
    - These perform poorly on facebook because it requires large amounts of sampling data
  + Faster process with smaller training dataset but poorer results
  + Researchers argue that both things perform similarly accuracy wise
  + Combining has better efficiency
  + Going for twitter is popular thanks to their api
  + the most common method uses in Lexicon based method is SentiWordnet and TF-IDF while for machine learning is Naïve Bayes and SVM
* Brief Summary of my Evaluation
  + Only goes to 2019
* Comparison to Other Works
* Bibliographic Details

**~~Paper 4 – A review: preprocessing techniques and data augmentation for sentiment analysis~~**

* ~~Main Themes~~
  + ~~Datasets~~
  + ~~Data preprocessing~~
  + ~~Data augmentation~~
* ~~Brief Content Summary~~
  + ~~Different types of formats: structured sentiments, unstructured, semi-structured~~
  + ~~Unstructured is the most difficult to work with and also what tweets are~~
  + ~~Traditional method is unsupervised learning~~
  + ~~Depends on size and quality of labelled datasets, scarce and not well labelled~~
  + ~~Texts such as tweets have a lot of noisy information, must clean the text~~
  + ~~Pre processing significantly improves the accuracy of classifiers~~
  + ~~Lots of data augmentation techniques~~
  + ~~Easy data augmentation (eda), easy to implement and improve the performance significantly~~
  + ~~Methods:~~
    - ~~Lowercase everything, classic technique~~
    - ~~Removing stop words (a, the, etc.), removed reduce dimensionality and improve performance cuz comp has to go through less~~
    - ~~Elongated character removal, (sooooooo -> so)~~
    - ~~Abbreviations or frequently misspelled words replaced~~
    - ~~Replacing emotional icons~~
    - ~~Removing punctuation~~
    - ~~Removing numbers (done after replacing emojis)~~
    - ~~POS part of speech handling, tagging every words as noun pronoun etc.~~
    - ~~Negation handling, replace not good by bad so as not to confuse (usually if a negation is followed by a positive word), however this does not work well~~
    - ~~Intensification handling (very, etc.), appending strong to it~~
  + ~~Augmentation only good for small datasets~~
  + ~~EDA techniques~~
    - ~~Synonym replacement~~
    - ~~Random swap, swap synonyms randomly~~
    - ~~Random delete~~
    - ~~Random insert~~
  + ~~They increase meaninglessness but they also increase accuracy~~
  + ~~Disadvantage: does not preserve meaning~~
  + ~~Back translator: translate to one language using google translate and then back again~~
  + ~~Will not achieve the same sentence but a close one~~
  + ~~Syntax tree transformations, syntactic parser builds syntax tree and then this is used to generate new sentence form, ex: moving from active to passive voice~~
  + ~~F-score of every dataset improved with pre processing techniques~~
  + ~~While other methods are more complicated to put in place, EDA techniques are easy and still improve performance well~~
  + ~~Works better on big datasets, must ensure they are large enough~~
* ~~Brief Summary of my Evaluation~~
  + ~~Does this for Vietnamese~~
* ~~Comparison to Other Works~~
* ~~Bibliographic Details~~

* + ~~<https://computationalsocialnetworks.springeropen.com/articles/10.1186/s40649-020-00080-x>~~

~~Paper 5 – EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks~~

* ~~Main Themes~~
  + ~~EDA~~
* ~~Brief Content Summary~~
  + ~~Four operations: synonym replacement, random insertion, random swap, random deletion~~
  + ~~EDA improves performance~~
  + ~~Particularly strong results for smaller datasets~~
  + ~~Across 5 datasets training with eda while using 50% of the available data resulted in the same performance as using normal training with all data~~
  + ~~ML and DL high accuracy on sentiment analysis~~
  + ~~While other methods of DA are valid they have high cost relative to performance gain~~
  + ~~For a given sentence, randomly chose and performed one of the EDA techniques~~
  + ~~Synonym replacement: randomly choose n words to replace with one of its synonym chosen at random~~
  + ~~Random insertion: random synonym of random word in sentence, put in random place in the sentence. Repeat n times~~
  + ~~Random swap: randomly choose 2 words in sentence and swap them. Repeat n times.~~
  + ~~Random deletion: randomly remove each word in the sentence with probability p~~
  + ~~Used RNNs and LSTM-RNN, CNNs are also good~~
  + ~~Improvement of 0.8% for full datasets and 3.0% for only 500 of the texts in them~~
  + ~~For the most part sentences re close to the original sentences and conserve the labels of their original sentences~~
  + ~~For synonym replacement if you replace too many words it makes things worse, probably because the sentence meaning is lost~~
  + ~~Alpha=0.1 seems to be the best~~
  + ~~Ntrain α naug 500 0.05 16 2,000 0.05 8 5,000 0.1 4 More 0.1 4 GET FIGURE~~
  + ~~LACK OF STANDARDIZED DATA AUGMENTATION IN NLP~~
* ~~Brief Summary of my Evaluation~~
  + ~~While it isn’t the best technique it’s the one I will be using as it works very well for small datasets and I will have small datasets because of the time and handyman constraint~~
  + ~~Is a better and cheaper alternative to contextual augmentation, noising, GAN, back-translation~~
* ~~Comparison to Other Works~~
  + ~~Paper 4 said the opposite, it works better on large datasets, they said that because you get more instances of misspelled words and slang to look at~~
  + ~~However this paper did not look specifically at text analysis~~
  + ~~And even with full larger datasets there was an improvement~~
* ~~Bibliographic Details~~

* + ~~<https://arxiv.org/abs/1901.11196>~~
  + ~~2019~~

**~~THIS PAPER HAS GITHUB AND EXPLANATIONS AT THE END OF IT~~**

~~Paper 6 – Role of Text Pre-Processing in Twitter Sentiment Analysis~~

* ~~Main Themes~~
  + ~~Why pre-processing matters~~
  + ~~Pre processing~~
* ~~Brief Content Summary~~
  + ~~Twitter data is very noisy~~
  + ~~This is a problem~~
  + ~~Graphical user interface, text

    Description automatically generated~~
  + ~~Everything ‘extra’ needs to be removed first: slang words, emoticons, stop words, punctuations, urls, etc.~~
  + ~~Split attached words~~
  + ~~The Natural Language Toolkit2 used to remove lemmatization and stop words~~
  + ~~Diagram

    Description automatically generated~~
  + ~~Algorithm that decides significance of a slang word based on ?~~
  + ~~Used SVM based classifier~~
  + ~~Applied normalization to the tweets~~
  + ~~Basically this is all about a technique to see whether or not their slang word is important and that it helps with classification~~
* ~~Brief Summary of my Evaluation~~
* ~~Comparison to Other Works~~
  + ~~They say that need to remove emoticons entirely, opposition to the previous works who state it is important for context~~
  + ~~Second paper that uses NL Toolkit2~~
* ~~Bibliographic Details~~

* + ~~<https://www.sciencedirect.com/science/article/pii/S1877050916311607>~~
  + ~~2016~~

~~Paper 7 – Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis~~

* ~~Main Themes~~
  + ~~Pre processing~~
  + ~~Twitter~~
  + ~~Sentiment analysis~~
* ~~Brief Content Summary~~
  + ~~Many papers ignore pre-processing step~~
  + ~~Accuracy and F1 measure improve when using methods of expanding acronyms and replacing negation but barely improves when removing stopwords, numbers or URLs~~
  + ~~Naïve bayes and random forest classifiers more sensitive than logistic regression and support vector machine classifiers when pre-processing methods were applied~~
  + ~~Saif et al. find that pre processing leads to significant reduction in vocabulary size (by 62%)~~
  + ~~As opposed to the last paper, Bao et al. find that accuracy increases when URLs are removed (????) maybe~~
  + ~~They agree with other stuff though~~
  + ~~They removed negations~~
  + ~~Removed URL links~~
  + ~~Replace repeat letters by three times their letter ex coooool -> coool to keep the sentiment~~
  + ~~Remove numbers~~
  + ~~Remove stop words~~
  + ~~Most researchers think that stop words are not necessary and there are previously made lists to use~~
  + ~~HAS LISTS OF STOP WORDS~~
  + ~~Expand acronyms~~
  + ~~an~~ *~~n~~*~~-gram model predicts x i {\displaystyle x\_{i}} X(i) based on x i − ( n − 1 ) , … , x i − 1 {\displaystyle x\_{i-(n-1)},\dots ,x\_{i-1}} X(i)-(n-1),………,X(i-1)~~
  + ~~word n grams are best for twitter sentiment analysis~~
  + ~~prior polarity score feature model is second feature model looked at~~
  + ~~prior polarity score is the sum of the sentiment score of each word in the tweet~~
  + ~~SenScore(w) = PMI(w, pos) − PMI(w, neg) Where w is a term in the lexicon, PMI(w, pos) is the PMI score between w and the positive class~~
  + ~~Popular supervised classifiers in literature:~~
    - ~~Support vector machine~~
    - ~~Naïve Bayes~~
    - ~~Logistic Regression~~
    - ~~Random Forest~~
  + ~~Stanford Twitter Sentiment Test dataset is quite small but widely used in literature for different evaluation tasks~~
  + ~~COULD POTENTIALLY USE IT FOR DATASET EXPANSION~~
  + ~~OR SemEval2014 dataset~~
  + ~~Or Stanford Twitter Sentiment Gold~~
  + ~~Or Sentiment strength Twitter dataset~~
  + ~~Or Sentiment Evaluation Dataset~~
  + ~~Table

    Description automatically generated~~
  + ~~They also find that URL removal does nothing with N-gram models~~
  + ~~In Prior polarity, it slightly reduces performance~~
  + ~~Okay so we DO want to remove URLs because while it does not help the classifier and in some cases even hinders it a bit, it’s impact is negligeable compared to the help in the process it will be, and after testing by randomly removing one word from each sentence it has a MUCH smaller impact than that, therefore is not equivalent to removing a word that could be helpful~~
  + ~~So removing stop words is good for N-gram models~~
  + ~~Removing numbers is effective preprocessing method, proven by doing same as URLs~~
  + ~~Removing stop words does nothing to N-gram models but does affect prior polarity just because the stop words have different sentiment polarity~~
  + ~~Removing repeated letters creates different results on all datasets~~
  + ~~Expanding acronyms helps~~
  + ~~So does replacing negation~~
* ~~Brief Summary of my Evaluation~~
  + ~~Numbers and stopwords barely changes things but they will still help performance and cost~~
  + ~~removing stop words, numbers, and URLs is appropriate to reduce noise but does not affect performance. Replacing negation is effective for sentiment analysis~~
* ~~Comparison to Other Works~~
* ~~Bibliographic Details~~

* + ~~<https://ieeexplore.ieee.org/document/7862202>~~
  + ~~2017~~

~~Paper 8 – A Comparison of Pre-processing Techniques for Twitter Sentiment Analysis~~

* ~~Main Themes~~
* ~~Brief Content Summary~~
  + ~~According to~~ [~~https://dl.acm.org/doi/pdf/10.1145/980972.981004?casa\_token=haFaPUJ3q34AAAAA:0EoSOhq-DvsGTQEPQdbMaogg0WRMAzSDlgd7tPPdAfPbRuyLqKzGcBfbBQOt3cGAyCFIR3\_vrzKM9Q~~](https://dl.acm.org/doi/pdf/10.1145/980972.981004?casa_token=haFaPUJ3q34AAAAA:0EoSOhq-DvsGTQEPQdbMaogg0WRMAzSDlgd7tPPdAfPbRuyLqKzGcBfbBQOt3cGAyCFIR3_vrzKM9Q) ~~, the total percentage of noise in a dataset reaches 40%~~
  + ~~Techniques are:~~
    - ~~Remove numbers,~~ [~~https://dl.acm.org/doi/10.1145/1645953.1646003~~](https://dl.acm.org/doi/10.1145/1645953.1646003) ~~argue that keeping them improve classification effectiveness~~
    - ~~Replace repetitions of punctuation, handled with representative tag~~
    - ~~Handling capitalized words~~
    - ~~Lowercasing~~
    - ~~Replace slang and abbreviations~~
    - ~~Replace elongated words~~
    - ~~Replace contractions~~
    - ~~Replace negations with antonyms~~
    - ~~Handling negations~~
    - ~~Remove stopwords~~
    - ~~Stemming, removing ending of words to get to their root form, generally produces good results~~
    - ~~Lemmatizing, pretty much the same thing~~
    - ~~Replace URLs and user mentions~~
    - ~~Spelling correction~~
    - ~~Remove punctuation~~
  + ~~Several datasets have been published, the most common will use positive, negative, neutral~~
  + ~~Reannotated the dataset~~
  + ~~Used Generalized Linear Model, from that family they chose Logistic Regression~~
    - ~~In this model, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function~~
  + ~~Also NB and SVM~~
    - ~~Naïve Bayes algorithms are the simplest probabilistic classification algorithms [~~[~~5~~](https://link.springer.com/chapter/10.1007/978-3-319-67008-9_31#ref-CR5)~~] that are widely used in sentiment analysis. They are based on the Bayes Theorem, which assumes a complete independence of variables. The Bernoulli algorithm is an alternative of Naïve Bayes, where each term is equal to 1 if it exists in the sentence and 0 if not. Its difference from Boolean Naïve Bayes is that it takes into account terms that do not appear in the sentence. It is a fast algorithm that deals well with high dimensionality.~~
    - ~~They try to find a set of hyperplanes that separate the space into dimensions representing classes. These hyperplanes are chosen in a way to maximize the distance from the nearest data point of each class. The Linear SVC is the simplest and fastest SVM algorithm assuming a linear separation between classes~~
  + ~~All models are linear~~
  + ~~Accuracy can be used as a good metric because the datasets are balanced~~
  + ~~Punctuation is important, removing it does not help~~
  + ~~Recommended preprocessing:~~
    - ~~Stemming, replacement of repetitions of punctuation, removing numbers~~
  + ~~Not recommended are:~~
    - ~~Removing punctuation, handling capitalized words, replacing slang, replcing negations with antonyms, and spelling correction~~
* ~~Brief Summary of my Evaluation~~
* ~~Comparison to Other Works~~
* ~~Bibliographic Details~~

* + ~~<https://link.springer.com/chapter/10.1007/978-3-319-67008-9_31>~~
  + ~~2017~~

~~Paper 9 – Evaluation Datasets for Twitter Sentiment Analysis A survey and a new dataset, the STS-Gold~~

* ~~Main Themes~~
  + ~~Dataset comparison~~
* ~~Brief Content Summary~~
  + ~~Comparing current datasets, they don’t have distinctive sentiment annotations among the entities within them~~
  + ~~STS-Gold tries to solve that problem by separating entities within a tweet~~
  + ~~The most common sentiment labels are positive, negative and neutral, but some evaluation datasets consider additional sentiment labels such as mixed, other or irrelevant~~
  + ~~some evaluation datasets also provide sentiment labels associated to targets (entities) within the tweets. However, these datasets do not distinguish between the sentiment label of the tweet and the sentiment labels of the entities contained within it~~
  + ~~purpose of this dataset is therefore to complement current state of the art datasets by providing entity sentiment labels, therefore supporting the evaluation of sentiment classification models at entity as well as tweet level.~~
  + ~~The Stanford Twitter sentiment corpus (http://help.sentiment140.com/), introduced by Go et al. [8] consists of two different sets, training and test. The training set contains 1.6 million tweets automatically labelled as positive or negative based on emotions. For example, a tweet is labelled as positive if it contains :), :-), : ), :D, or =) and is labelled as negative if it contains :(, :-(, or : (. Although automatic sentiment annotation of tweets using emoticons is fast, its accuracy is arguable because emoticons might not reflect the actual sentiment of tweets~~
  + ~~STS Test set is relatively small and used for evaluation tests~~
* ~~Brief Summary of my Evaluation~~
* ~~Comparison to Other Works~~
* ~~Bibliographic Details~~

* + ~~<http://ceur-ws.org/Vol-1096/paper1.pdf>~~

~~Paper 10 – Atalaya at TASS 2019: Data Augmentation and Robust Embeddings for Sentiment Analysis~~

* ~~Main Themes~~
  + ~~Data augmentation~~
  + ~~And other stuff~~
  + ~~Spanish tweets~~
* ~~Brief Content Summary~~
  + ~~Used negation handling ie find negation words and add prefix NOT to following tokens~~
* ~~Brief Summary of my Evaluation~~
  + ~~Applied to Spanish tweets~~
* ~~Comparison to Other Works~~
* ~~Bibliographic Details~~
  + [~~https://arxiv.org/abs/1909.11241~~](https://arxiv.org/abs/1909.11241)
  + ~~2019~~

Paper 11 – Emotion Detection and Analysis on Social Media

* Main Themes
  + NLP and ML
  + Classifying text on basis of emotions
* Brief Content Summary
  + Six different Emotion-Categories: Happiness, Sadness, Fear, Anger, Surprise and Disgust
  + Today most SA is that it only informs about negative, positive, neutral reactions
  + TWEEPY TO COLLECT TWEETS, python library
  + Created bag of words thing for emotional words
  + Created a set of degree words (50)
* Brief Summary of my Evaluation
* Comparison to Other Works
* Bibliographic Details

* + <https://arxiv.org/pdf/1901.08458.pdf>
  + 2018

Paper 13 – Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset

* Main Themes
  + Support vector machine and naïve bayes comparison
* Brief Content Summary
  + Tested on airline reviews
  + SVM performed way better than naïve bayes
  + They pre-processed with known techniques
  + Negative, neutral, positive
  + SVM had better scores in Accuracy and precision and worse scores in recall and F1, wayyy better in accuracy though, almost 6%
* Brief Summary of my Evaluation
  + Very boldly claim that sentiment analysis is the most used research topic
* Comparison to Other Works
* Bibliographic Details
  + <https://ieeexplore.ieee.org/abstract/document/9117512?casa_token=fp-57EeRlXwAAAAA:gLkjGY-jjmQ9Xj1BRUWGSwksADk5RG0gqkthIafC9GmYwgWlkws5HSVVs4YNVRpajZJy8ziFxg0>
  + 2019

Paper 14 – A Study on Sentiment Analysis Techniques of Twitter Data

* Main Themes
  + Sentiment analysis ML methods
* Brief Content Summary
  + Sentiment analysis can be performed at 3 levels:
    - Document level: entire docu is seen as positive or negative
    - Sentence level: sentence
    - level, sentences/documents can be categorized as “positive”, “negative” or “non-partisan” in light of certain aspects of sentences/archives and commonly known as “perspective-level assessment grouping”.
  + Machine learning classifiers: naïve bayes, maximum entropy, support vector machine
  + Naïve bayes widely used in task of classifying texts
  + Maximum entropy:
  + SVM is known to perform well in sentiment analysis, <https://ieeexplore.ieee.org/document/7951748> they showed that
  + Four basic twitter sentiment analysis approaches: supervised ML based, ensemble methods, lexicon based, hybrid
  + Many different papers find different things for which of the three (SVM, MaxEnt, NB) worked the best in supervised learning approach
  + Ensemble methods combine multiple classifiers to get more precise and accurate predictions, these methods might be useful for improving classification accuracy of twitter posts
  + Ensemble seems to be overall more effective than just supervised ml
  + Unigram based SVM is normally considered benchmark against which other strategies are compared
  + Integrating multiple features improves classification accuracy
  + Apparently incorporating semantic with unigram features produces better performance than baseline feature selection
  + Research outcomes demonstrated that machine learning techniques; for example, the SVM and MNB produced the greatest precision, especially when multiple features were included
  + Machine learning algorithms, such as The Naive Bayes, Maximum Entropy, and SVM, achieved an accuracy of approximately 80% when n-gram and bigram model were utilized.
  + Ensemble and hybrid-based Twitter sentiment analysis algorithms tended to perform better than supervised machine learning techniques, as they were able to achieve a classification accuracy of approximately 85%
  + In general, it was expected that ensemble Twitter sentiment-analysis methods would perform better than supervised machine learning algorithms, as they combined multiple classifiers and occasionally various features models. However, hybrid methods also performed well and obtained reasonable classification accuracy scores, since they were able to take advantage of both machine learning classifiers and lexicon-based Twitter sentiment-analysis approaches.
* Brief Summary of my Evaluation
* Comparison to Other Works
* Bibliographic Details
  + <https://www.researchgate.net/profile/Abdullah-Alsaeedi/publication/331411860_A_Study_on_Sentiment_Analysis_Techniques_of_Twitter_Data/links/5c78175ba6fdcc4715a3d664/A-Study-on-Sentiment-Analysis-Techniques-of-Twitter-Data.pdf>
  + 2019

Paper 15 – A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis

* Main Themes
  + Lexicon based approaches
  + Movie reviews
* Brief Content Summary
  + There are two main approaches to extract sentiment: lexicon based and machine learning approach
  + Lexicon based requires predefined lexicon while machine learning automatically classifies using training data
  + Lexicon approaches do not use training data
  + Problem with ML approaches: since it is trained on a limited data it might struggle to classify out of the dataset it was trained on, however this has been considered by earlier researchers and determined that larger datasets can help that problem
  + Parts of speech tagging
  + NLTK, NLP platform for python
  + This pre processes reviews, then performs POS tagging, then extracts positive and negative words, matches them with sentiment score, uses sentiment polarity and determines score
  + Textblob: python library
  + Sentence level analysis, will look at sentiment of each sentence and determine a score
  + Vader is lexicon and rule based, pretty much the same as the others but it tells how positive or negative comment is as well
  + Vader had the best precision, f score, and accuracy
* Brief Summary of my Evaluation
* Comparison to Other Works
* Bibliographic Details

* + <https://www.researchgate.net/profile/Naulegari-Janardhan/publication/333602124_A_Comprehensive_Study_on_Lexicon_Based_Approaches_for_Sentiment_Analysis/links/5d13452ca6fdcc2462a688ed/A-Comprehensive-Study-on-Lexicon-Based-Approaches-for-Sentiment-Analysis.pdf>
  + 2019

Paper 16 – Lsislif: Feature Extraction and Label Weighting for Sentiment Analysis in Twitter

* Main Themes
* Brief Content Summary
  + Three types of approaches: lexicon, machine learning, hybrid
  + hybrid achieves the best performance
  + Mohammad et al. showed that using lexicons with SVM showed the biggest performance increase
  + Hamdan et al. showed lexicon-based classifiers helped with logistic regression as well
  + Hybrid methods seem to be the way to go here
  + Unigram and bigrams extracted from feature space
* Brief Summary of my Evaluation
* Comparison to Other Works
  + Says that limitations of lexicon approaches are limited size of lexicons and that the lexicons are difficult to build
* Bibliographic Details

* + <https://aclanthology.org/S15-2095.pdf>
  + 2015

Paper 18 – Sentiment Classification: Review of Text Vectorization Methods: Bag of Words, Tf-Idf, Word2vec and Doc2vec

* Main Themes
  + Text vectorization methods
* Brief Content Summary
  + Machine learning algorithms work with vectorization because of the machines needing to understand what is happening
  + The commonly used ones are bag of words, TF-IDF, word2vec, and doc2vec
  + Vectorization the transformation or encoding of texts into numerical vectors for machine learning
  + Term frequency inverse document frequency is widely used because it reveals relative importance of words in corpus
  + Word2vec: addresses problem that TFIDF has of semantic relatedness to words
  + Many types of sentiment analysis and text classification tasks can use these word vectors
  + Doc2vec just adaptation of word2vec to accommodate for sentences etc.
  + Doc2vec sidesteps most of the weaknesses of bag of words by representing sequence of text with fixed length feature vector while retaining order and semantics of words
  + Bag of words: treats doc as collection of words regardless of the grammar order, treats each docu as a vector with fixed length (usually length of corpus)
  + TFIDF: scaled down version of bag of words, This means multiplying the number of times a word appears in a review text by the logarithm of dividing the total number of reviews in the corpus by the number of reviews in which the term appears when considering review texts as documents
  + Word2vec: It's a distributed vector representation of words based on the assumption that words with comparable meanings in the same context are represented similarly. As a result, word vectors with similar meanings are clustered together in the vector space. The challenges of high dimensionality and loss of word context that are unique to the bag of words and its n-gram versions are addressed by this family of vectorization approaches.
  + CBOW: word2vec architecture that can predict current or target words based on previous words
  + result of this study shows that TF-IDF feature vector representations generally outperforms other two(2) vectorization methods word2vec and doc2vec, specifically in book review sentiment classification. This can be validated based on the study conducted by (Haisal A. D et al 2021) on the topic: A Scheme of Pairwise Feature Combinations to Improve Sentiment Classification Using Book Review Dataset.
  + combination of TF-IDF word2vec performed best compared to all other methods either combined or singly.
* Brief Summary of my Evaluation
* Comparison to Other Works
* Bibliographic Details

* + <https://slujst.com.ng/index.php/jst/article/view/266/115>
  + 2022

Paper 20 – Identification of Potential Cyber Bullying Tweets using Hybrid Approach in Sentiment Analysis

* Main Themes
* Brief Content Summary
  + Lexicon based approaches do not need training data and are quite good at sentiment polarity
  + They use hybrid approach, three main steps
  + Knowledge based sentiment analysis, results reinforced with ML
  + Step 1: build dataset, segment it into emoticons and text, processed separately
  + Step 2: after emoticons compiling, assign individual polarity to each emoticon, then overall polarity of all emoticons in one tweet is calculated through aggregation and fed back to the network
  + Step 3: text data processed separately through 2 pronged system, lexicon and Machine Learning approach (NB and Linear support vector classifier SVC)
  + Step 4: obtain dual polarity through the two ML methods
  + Step 5: polarity aggregation, using two pronged strategy
  + Then find highly negative tweets and use POS to identify if it’s aimed at a person
  + If yes it’s a risk
  + So three polarities for each sentence
  + Accuracy of classification at 70,3%
* Brief Summary of my Evaluation
* Comparison to Other Works
* Bibliographic Details

* + <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9001476&tag=1>
  + 2018

Paper 21 – Sentiment Analysis of Cyberbullying on Instagram User Comments

* Main Themes
  + Cyberbullying detection on instagram
  + Naïve bayes
  + TD-IDF feature extraction
  + K-Fold Cross Validation for testing
* Brief Content Summary
  + By complexity, NB is simpler and more conventional (?) than others, computational times will be shorter
  + Two classes: yes cyberbullying and no
  + Pre-process
  + Feature extraction:
    - TF-IDF, no explanation of why
    - Find number of terms in each document (TF), calculate number of documents that have that term (DF), IDF calculation idk how, then multiply TD by IDF
  + K-fold validation evaluation
  + Dataset randomly divided into number of K pieces, each experiment uses K partition data as testing and rest as data training and repeated with all K partitions
  + So at 84% accuracy, and stemming does help
* Brief Summary of my Evaluation
  + Did this in another language
  + Do not explain how they identify strongly negative comments compared to just negative comments, can guess through other works that the TF-IDF method did that
* Comparison to Other Works
* Bibliographic Details
  + <http://commdis.telkomuniversity.ac.id/jdsa/index.php/jdsa/article/view/20/7>
  + 2019

Paper 22 – Social media sentiment analysis: lexicon versus machine learning

* Main Themes
  + Lexicon versus machine learning
  + And both too
* Brief Content Summary
  + Both approaches perform similarly
  + Work better when classifying positive than negative
  + Combined approach does significantly better with identifying positive
  + Further research is needed to improve accuracy of negative tweet detection
  + Two big approaches: classification using lexicon of weighted words, widely used as does not require preprocessing
  + Machine learning reported to be more accurate
  + Are these two existing sentiment analysis techniques appropriate for the analysis of social media conversations?
  + To what extent do the results from the two approaches differ when used on social media conversations?
  + Does a combined approach improve the overall accuracy of the sentiment classification of social media conversations?
  + Lexicon based approach relies on dictionary of opinion words
  + Standard lexicon like LIWC include sentiment dictionary
  + Drawback to using lexicon based: polarity classification can vary, like ‘unpredictable’ can mean good or bad depending on context
  + Machine learning trains sentiment classifiers using unigrams or bigrams usually
  + High quality labelling is expensive and time consuming and small datasets lead to poor results
  + Look at some studies that use lexicons to label data and then use ML on that data, but still not as accurate as using manually labelled data
  + Half as testing set half as training set
  + Used two different ML algos to classify positive and then negative tweets
  + ML algos: maxent and bagging
* Brief Summary of my Evaluation
  + 850 consumer comments from 83 facebook brand pages
  + They are the only ones who have had less success with negative tweets than positive ones, possibly because they used two different algorithms for each one for some reason
  + They did this on ‘consumer generated content’ we are not looking at consumers
  + Good methodology however the size of the dataset was not huge
  + Two positive or negative
  + They also say just using both and combining their results is good
* Comparison to Other Works
* Bibliographic Details
  + <https://www.emerald.com/insight/content/doi/10.1108/JCM-03-2017-2141/full/html>
  + 2017

Paper 22 – A Comparative Study on Vectorization and Classification Techniques in Sentiment Analysis to Classify Student-Lecturer Comments

* Main Themes
* Brief Content Summary
  + Very big dataset, 52 000 comments
  + Random forest worked best
  + Started with a three class model and then moved to 5 class model!!
  + TF-IDF performed better than Count (Binary) vectorization
  + Machine learning different because textual data has to be converted to numeric first
  + Count vectorization has shortcomings that are fixed by TF-IDF that assigns weights to words
  + 5 class classification
  + They go for NB, Decision Tree, Random Forest, and SVM
  + State that 3 class datasets are mainly used in research, this is supported by our findings
  + How to measure how good it is, Precision (PR), Recall (RC), and Accuracy (ACC)
  + They pre-process everything first
  + Then used TF-IDF
  + Classify comments by score, Admiration given +2, Complete disappointment -2, etc.
  + Random forest seemed more accurate compared to other classifiers: SVM, gradient boosting, SVM, and NB
  + They conclude that TF-IDF can be used for effective with student comments
* Brief Summary of my Evaluation
  + Big dataset
  + The point of the paper was to determine whether or not sentiment analysis was a good way to evaluate course and lecturers, however they talk about sentiment analysis methods
  + Good methodology
  + The 5 class one is slightly less accurate this makes sense because there is more space for objectivity and being wrong
  + Did not test bigrams or trigrams
* Comparison to Other Works
* Bibliographic Details
  + <https://reader.elsevier.com/reader/sd/pii/S1877050920323954?token=C705F0E4FC3E997465DA0ADC3F1DDA5DB95513EBCBA17699564279DC8D76E9E98AC158423B4D456603C9D58BEF858334&originRegion=eu-west-1&originCreation=20221028130524>
  + 2020

Paper 23 – Sentiment Analysis-Based Method to Prevent Cyber Bullying

* Main Themes
  + Cyber bullying
  + RNN
* Brief Content Summary
  + Opinion mining and sentiment analysis different
  + Data preprocessing first
  + 1000 messages were collected
  + Tokenize/word segmentation
  + STOPWORDS: they consume a lot of computational resources and do not add any semantic value to the text
* Brief Summary of my Evaluation
  + They go after RNN which we will not be using to limit the scope of our project here
  + Not huge dataset
* Comparison to Other Works
  + Also uses pos, neg,neutral
  + Used accuracy, not the best for evaluation as it doesn’t work well if there are different amounts of data in each class as there usually is
* Bibliographic Details
  + <https://link.springer.com/chapter/10.1007/978-981-19-2456-9_73>
  + 2021

Paper 25 – A Comparative Analysis of Machine Learning Classifiers for Twitter Sentiment Analysis

* Main Themes
* Brief Content Summary
  + Look at unigrams and bigrams as feature spaces
  + Higher n grams not good for twitter cuz some tweets only have a few words
  + Some researchers do stemming before classification
  + Lexicon based and machine learning based
  + Major pitfall of lexicon: no mechanism to deal with context
  + Naïve bayes, attributes/features are assumed to be independent of each other which makes naïve assumption
  + STRENGTHS: handling noisy data well, null values ignored and irrelevant features uniformly distributed so they don’t have much influence on classification result
  + WEAKNESS: there is a possibility of a probability that is calculated at 0 which will ruin the whole probability thing because it gets multiplied with everything else
  + There are other Naïve Bayes models that can help with that
  + Multinomial Naïve Bayes Classifiers, Bernoulli Naïve Bayes classifiers
  + SVM is non-probabilistic
  + Sequential Minimal Optimization (SMO), used for training support vector machines
  + It breaks the SVM optimization problem into a series of smallest possible sub problems
  + Comparing multinomial NB, Bernoulli NB, and SVM
  + These are the most commonly used classifiers in the literature
  + Pre process using acceptable stuff
  + Unigrams + bigrams as feature spaces, bigrams make larger feature spaces
  + They end up with:
    - Unigram with term polarity.
    - Unigram with Term Frequency.
    - Bigrams with term polarity.
    - Bigrams with term Frequency
  + Term frequency: number of occurrences in the document – this is used with TF-IDF
  + Term polarity: occurrence or absence of a term in the document regardless of how often it happened – this is used with BoW
  + Unigrams dataset higher accuracy overall
  + And training time is shorter
  + MNB had the best classification results with frequency thing
  + Naïve bayes classifier outperformed others
  + Unigrams are better specifically for twitter
  + Did not know if frequency or polarity is better
  + SVM took the longest processing time AND had least accurate results
* Brief Summary of my Evaluation
  + Good literature review/survey, collects many papers, however from 2016 so maybe not the most up to date
  + They did this on twitter data, might be important for accuracy of results
  + Used STS dataset with only 369 tweets, and collected in 2009
* Comparison to Other Works
* Bibliographic Details
  + <file:///C:/Users/cleme/Downloads/AComparativeAnalysisofMachineLearningClassifiersforTwitterSentimentAnalysis.pdf>
  + 2016

Paper 26 – A Comparative Study of Sentiment Analysis Using NLP and Different Machine Learning Techniques on US Airline Twitter Data

* Main Themes
  + NLP
  + Different ML techniques
  + Airline Twitter data
* Brief Content Summary
  + NLP techniques: BoW, TF-IDF
  + ML techniques: SVM, Logistic Regression, Multinomial NB, Random Forest
  + Our best approaches provide 77% accuracy using Support Vector Machine and Logistic Regression with Bag-of-Words technique.
  + First they applied NLP techniques to pre-process and vectorize the data
  + Then applied SVM, Multinomial NB, Random Forest, and LR
  + NLP techniques are BoW and TF-IDF
  + Methodology: Collect dataset to train and test ML Classifier
  + Pre-process the dataset
  + Convert textual data into vector form using NLP techniques
  + Dividing the dataset into training and testing groups and train the ML
  + Table

    Description automatically generatedHighly negative dataset
  + The idea behind BoW is to mark the occurrence of the word in each tweet from the vocabulary to convert it into a vector representation. We should use 1s and 0s to mark the appearance of each of these words.
  + (TFIDF): It is used to find the important terms or words that appear in the document or tweet based on their frequency. In TF-IDF, the less frequent word means more important.
  + Measure with accuracy, precision, recall, and F-score matrices, they weighted the average of precision, recall, and F-score
  + SVM and LR provide highest accuracy with 77%
  + Based on their experiments the best approach is SVM or logistic regression with the BoW technique
* Brief Summary of my Evaluation
  + Focusses on one topic but uses tweets still so better than nothing
  + Use it for customer satisfaction specifically
  + Large, unbalanced and multi-class dataset
  + Not much information on implementation of methodology
  + Say their approach is better than other studies but what did they do differently
* Comparison to Other Works
* Bibliographic Details
  + <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9641336>
  + 2021