History of Automatic Sarcasm Detection Systems

Prepared by: Hari Thapliyal
Reviewed by: Dr. Anil Vuppala.
Date: 27-Aug-20

1. Introduction

This document helps any person who is interested in knowing a brief history of sarcasm detection in the English and Hindi language text. We are focused on automatic sarcasm detection. Different approaches are used by various researchers over the period of last 20 years. Recently we see surge of machine learning algorithms, neural networks, and transformers in sarcasm detection. In this document we are trying to analyze features used, algorithms used and results achieved by different researchers.

2. The Journey

In 2002, (Turney, 2002) in their work "The perfect solution for detecting sarcasm in tweets #not" presents an unsupervised learning-based algorithm for classification of review in English language. There dataset was opinion survey of product. They used Semantic Orientation (SO) to perform this work. SO of a phrase is calculated using adverbs and adjectives used in the phrase. The experiments were done for text of various domains like automobiles, banks, movie review and travels. The results of this experiment vary from domain to domain between 66 to 84%. The power of this SO in Hinglish language sarcasm detection can used and verified.

In "Clues for detecting irony in user-generated contents" (Carvalho et al., 2009) used emoticons, heavy punctuation marks, quotation marks and positive interjections, onomatopoeic expressions for laughter to predict the sarcasm. They concluded that if sentence has more laughter related feature than there are high chances it will be ironic. But heavy use of puncution and quote is also sign of sentence being sarcastic.

In "Semi-supervised Recognition of Sarcastic Sentences in Twitter and Amazon" (Davidov et al., 2010) used English language dataset which has twitter and amazon reviews. They created features based on Meta Tag (User, Company, Product, Title, Author), Link, HashTags,

Punctuation. They used KNN classifier and they found F1 on amazon data is 78% and F1 on twitter data is 83%.

In "Identifying Sarcasm in Twitter: A Closer Look." (González-Ibáñez et al., 2011) used English Tweets and found Accuracy with SVM classifier varies between 55.59% to 75.78% depending upon tweet format. They used following features.

- Lexical Features: unigram, dictionary based (Linguistic Processes (e.g., adverbs, pronouns), Psychological Processes (e.g., positive and negative emotions), Personal Concerns (e.g, work, achievement), and Spoken Categories (e.g., assent, non-fluencies)) + WordNet Affect + interjection + punctuation
- Pragmatic Features: positive emotions like smily, negative emotions like frowning face. ToUser like @Name
- $\chi 2$ test to select features

In "Baselines and bigrams: Simple, good sentiment and topic classification" (Wang and Manning, 2012) claims that although SVM and NB both are good for text classification. However, NB does better than SVM when document is small size but in case document size is bigger than SVM performs better than NB.

(Liebrecht et al., 2013) demonstrated a sarcasm detection system in "The perfect solution for detecting sarcasm in tweets #not". This system was developed for tweets in Dutch language. They used 78,000 sarcastic tweets, along with normal tweets dataset, while adding normal tweet ensured that none of the normal tweet is part of sarcastic dataset. Split the sarcastic tweets into train-test and added with normal tweet into train dataset to train the model. Then test the model using test dataset which has only sarcastic tweets. There experiments leads to AUC of .79. This paper gives an overall approach of building sarcasm detection system in other than English language. But it does not address the problem which Hinglish language has. There test train split and model training approach looks good for non-English language.

(Asghar et al., 2014) developed a system to detect negative, positive, and neutral sentiments for English language tweets. In their work "Lexicon-Based Sentiment Analysis in the Social Web" they claimed their system can detect and score the slang used in the tweet. This system has Accuracy of 92% for binary classification and 87% for multinomial classification. An approach to get tweets clean text is discussed for English language tweets.

In "Sparse, contextually informed models for irony detection: Exploiting user communities, entities and sentiment." (Wallace et al., 2015) claimed that capturing previous and following comments on Reddit improves classification performance. They used POS features like NNP, sentiment, subreddit (like progressive or conservative; atheism or Christianity). They used reddit irony corpus.

In "Sarcasm Detection on Twitter: A Behavioral Modeling Approach." (Rajadesingan et al., 2015) mentioned that Users behavioral information is also beneficial as it captures useful contextual information in Twitter post. They created 335 SCUBA in following categories Sentiment Score, Sentiment Transition between past and present, Sarcasm as a complex form of expression, emotion (mood, frustration, affects and sentiments), language familiarity, sarcasm familiarity, environment familiarity, written expression related, structural variation. With logistic regression they achieved highest accuracy of 83.46%.

In "Figurative messages and affect in twitter: Differences between# irony,# sarcasm and# not." (Sulis et al., 2016) evaluated novel set of sentiment, structural and psycholinguistic features. They used different lexical resources developed for English language. They used classification algorithms NB, DT, RF, LR, SVM. They used english language resources like HL, GI, DAL, ANEW, SS, EmoSN, EmoLex, SN, LIWC, EWN, SWN, AFINN. They found Irony shows highest experience of joy, anticipation and trust. It shows least behaviour of sadness, fear, surprise, disguest, anger.

In "Irony detection in twitter: The role of affective content" (Fa ́rias et al., 2016) presented the performance of knn classifier which was fed with a feature set that captures a wide range of linguistic phenomena like structural, emotional. They used SVM, DT, NB classifiers and achieved accuracy between 73-96% depends upon datasets and classifier used.

In "An Empirical, Quantitative Analysis of the Differences between Sarcasm and Irony" (Ling and Klinger, 2016) explored syntactic and sentiment related features. They demostrated that separating sarcasm from normal text with 90% accuracy but separting irony from sarcasm achieved only 79% accuracy. In this experiement they used English language twitter data.

In "Unsupervised irony detection: A probabilistic model with word embeddings" (Nozza et al., 2016) introduced a A novel unsupervised probabilistic modeling approach to detect irony. They developed a Topic Irony Model (TIM) and used word embedding to develop a unsupervised

irony detection system. This experiment was conducted on education, politics and humor domain. F1 score achieved using TIM+WE system is between 82.92 to 88.34% depending upon domain.

Sarcasm is the major factor which can flip the meaning of a written or spoken phrase. To avoid the negativity people use positive words to communicate negative message, (Desai and Dave, 2016). In their work "Sarcasm detection in hindi sentences using support vector machine" they have used libsvm algorithm for multiclass classification. This paper uses 5 grades of sarcasm Non-Sarcastic, Mild Positive Sarcastic, Extreme Positive Sarcastic, Mild Negative Sarcastic and Extreme Negative Sarcastic. This work demonstrates the accuracy between 60% to 84% depending upon, whether sentence has any clue like emoticon, tag etc of sarcasm. This work suggests usage of lexical, pragmatic, and lingustic features along with emoticons, hashtag, punctuation marks to detect the sarcasm.

In "Harnessing Online News for Sarcasm Detection in Hindi Tweets" (Bharti et al., 2017) mentions that sarcasm detection is one of the most complex work in Hindi Language and the reason for that is words in Hindi language are rich in morphology. This paper discusses a system to sarcasm detection in Hindi tweet but for that it is taking help of online news related to the tweet. This work demonstrates accuracy of 70.4%

In "A novel automatic satire and irony detection using ensembled feature selection and data mining" (Ravi and Ravi, 2017) used Linguistic, Semantic, Psychological, unigram features. They further created group of features like LIWC features (L), TAALES Features(T), Unigram Features(D), then created ensemble feature subset. To select the useful feature they used IG, GR, Chi, CORR, TSTAT. They used English language text from Newswire, Satire news articles and Amazon. Experimenters used SVM (Liner, RBF, Sigmoid, Polynomial), LMT, LR, RF, NB, BN, MLP classifiers. They achieved the highest F1 score 96.58% with (L+T+D features) + GR feature selector + SVM RBF Classifier

In "Exploring the fine-grained analysis and automatic detection of irony on twitter" (Van Hee et al., 2018) used SVM classifier in combination of lexical, semantic and syntactic features passed through an SVM classifier and it outperformed LSTM deep neural network approaches.

(Potamias et al., 2020) published their work "A Transformer-based approach to Irony and Sarcasm detection". In this paper they mentioned figurative language (FL) is ubiqutious and irony and sarcasm detection is challenging. They observed that social media users violates

grammar rules and heavily use figurative language to communicate. Author states that classical machine learning algorithms such as k-Nearest Neighbors (KNN), SVM, and tree-based models (Decision Trees, Random Forest) are inappropriate for real world applications, due to their demand of hand-crafted feature and exhaustive pre-processing to make text clean for performing any NLP task. To develop a reasonable KNN or SVM based model, there should be a lot of effort to embed sentences on word level to a higher space. After this only a classifier may recognize some patterns. They used Deep Learning methodologies for sarcasm detection. They conducted various experiments using ELMo, FastText, XLNet, BERT-cased, BERT-uncased, RoBERTa, UPF, ClaC, DESC, USE and NBSVM.

As per (Zhang et al., 2020) in their work "Semantics-aware BERT for Language Understanding" existing language representation models like ELMo, GPT and BERT exploit plain context-sensitive features such as character or word embeddings. These models rarely consider incorporating structured semantic information which can provide rich semantics for language representation. Authors proposed semantics-aware BERT and claims it is a simple in compare to BERT and more powerful. SemBERT (large) could predict the sentiment with accuracy as 94.5% on SST2. This looks impressive results however this limited to English language. In the absence of sufficient corpus size in Hinglish we cannot perform similar experiment using BERT.

In their work "Multi-Rule Based Ensemble Feature Selection Model for Sarcasm Type Detection in Twitter" (Sundararajan and Palanisamy, 2020) attempted to detect the sarcasm and further classify sarcasm into four categories namely rage, rude, polite and deadpan. Authors exploited a twitter dataset for this work. They extracted four types of features Lexical, Intensifiers, Pragmatic, Uppercase words. They extracted following 20 features from this tweet dataset: 1. noun count, 2. verb count, 3. positive intensifier, 4. negative intensifier, 5. bigram, 6. trigram, 7. skip gram, 8. unigram, 9. emoji sentiment, 10. sentiment score, 11. interjections, 12. punctuators, 13. exclamations, 14. question mark, 15. uppercase, 16. repeat words count, 17. positive word frequency, 18. negative word frequency, 19. polarity flip, and 20. parts of speech tagging. Using these features authors developed 8 models using following 8 algorithms 1. Random Forest, 2. Naive Bayes, 3. Support Vector Machine (SVM), 4. K-Nearest Neighbor (KNN), 5. Gradient Boosting (GBC), 6. AdaBoost, 7. Logistic Regression, and 8. Decision Tree. Their final goal is not to classify tweet as sarcastic or non-sarcastic tweet. They wanted to classify sarcastic tweet into four different categories. They reported accuracy of sarcasm

detection as 92.7% and accuracy of classification of sarcastic tweet among these four categories varies between 86.61% to 99.79% depending up type of emotion.

For categorization of the sarcastic tweet first they deployed certain rules to create linguistic features (L), sentiment features (S), contradictory features (C). Following this they used different combination of these features to create different ensembles models.

Authors obtained tweets through Twitter API (Tweepy and Twython) on the basis of the following hash tags: #sarcasm, #sarcastic, #Sarcasm, and #notSarcasm. A total of 76,799 tweets are collected, tweets that are non-English and non-Roman script are removed. During the pre-processing steps they removed any URL, Retweet, Link related text from the tweet.

3. References

[1.] Asghar, M.Z., Kundi, F.M., Khan, A. and Ahmad, S., (2014) Lexicon-Based Sentiment Analysis in the Social Web. *J. Basic. Appl. Sci. Res*, 46, pp.238–248.

[2.] Bharti, S.K., Sathya Babu, K. and Jena, S.K., (2017) Harnessing Online News for Sarcasm Detection in Hindi Tweets. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10597 LNCS, pp.679–686.

[3.] Carvalho, P., Sarmento, L., Silva, M.J. and De Oliveira, E., (2009) Clues for detecting irony in user-generated contents: Oh...!! it's 'so easy' ;-). *International Conference on Information and Knowledge Management, Proceedings*, pp.53–56.

[4.] Davidov, D., Tsur, O. and Rappoport, A., (2010) Semi-supervised recognition of sarcastic sentences in twitter and Amazon. *CoNLL 2010 - Fourteenth Conference on Computational Natural Language Learning, Proceedings of the Conference*, July, pp.107–116.

[5.] Fańias, D.I.H., Patti, V. and Rosso, P., (2016) Irony detection in twitter: The role of affective content. *ACM Transactions on Internet Technology*, [online] 163, pp.1–24. Available at: http://dx.doi.org/10.1145/2930663 [Accessed 17 Aug. 2020].

[6.] González-Ibáñez, R., Muresan, S. and Wacholder, N., (2011) Identifying Sarcasm in Twitter: A Closer Look. In: *Proceedings ofthe 49th Annual Meeting ofthe Association for Computational Linguistics:shortpapers*. [online] pp.581–586. Available at: http://www.vidarholen.net/contents/interjections/ [Accessed 16 Aug. 2020].

[7.] Van Hee, C., Lefever, E. and Hoste, V., (2018) Exploring the fine-grained analysis and automatic detection of irony on Twitter. *Language Resources and Evaluation*, 523, pp.707–731.

[8.] Liebrecht, C., Kunneman, F. and Bosch, A. Van den, (2013) The perfect solution for detecting sarcasm in tweets #not. [online] June, pp.29–37. Available at: http://www.aclweb.org/anthology/W13-1605.

[9.] Ling, J. and Klinger, R., (2016) An empirical, quantitative analysis of the differences between sarcasm and Irony. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. [online] pp.203–216. Available at: http://dx.doi.org/10.1007/978-3-319-47602-5_39 [Accessed 16 Aug. 2020].

[10.] Nozza, D., Fersini, E. and Messina, E., (2016) Unsupervised Irony Detection: A Probabilistic Model with Word Embeddings. In: *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*. [online] SCITEPRESS - Science and Technology Publications, pp.68–76. Available at: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006052000680076 [Accessed 16 Aug. 2020].
[11.] Potamias, R.A., Siolas, G. and Stafylopatis, A.G., (2020) A transformer-based approach to irony and sarcasm detection. *Neural Computing and Applications*.
[12.] Rajadesingan, A., Zafarani, R. and Liu, H., (2015) Sarcasm detection on twitter:A behavioral modeling approach. *WSDM 2015 - Proceedings of the 8th ACM International Conference on Web Search and Data Mining*, pp.97–106.
[13.] Ravi, K. and Ravi, V., (2017) A novel automatic satire and irony detection using ensembled feature selection and data mining. *Knowledge-Based Systems*, [online] 120, pp.15–33. Available at: http://dx.doi.org/10.1016/j.knosys.2016.12.018.
[14.] Sulis, E., Irazú Hernández Farías, D., Rosso, P., Patti, V. and Ruffo, G., (2016) Figurative messages and affect in Twitter: Differences between #irony, #sarcasm and #not. *Knowledge-Based Systems*, [online] 108, pp.132–143. Available at: http://dx.doi.org/10.1016/j.knosys.2016.05.035 [Accessed 16 Aug. 2020].
[15.] Sundararajan, K. and Palanisamy, A., (2020) Multi-rule based ensemble feature selection model for sarcasm type detection in Twitter. *Computational Intelligence and Neuroscience*, 2020.
[16.] Turney, P.D., (2002) Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. [online] July, pp.417–424. Available at: http://arxiv.org/abs/cs/0212032.
[17.] Wallace, B.C., Choe, D.K. and Charniak, E., (2015) Sparse, contextually informed models for irony detection: Exploiting user communities, entities and sentiment. In: *ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference*. [online] pp.1035–1044. Available at: http://www.reddit.com [Accessed 16 Aug. 2020].
[18.] Wang, S. and Manning, C.D., (2012) Baselines and bigrams: Simple, good sentiment and topic classification. *50th Annual Meeting of the Association for Computational Linguistics, ACL 2012 - Proceedings of the Conference*, 2July, pp.90–94.
[19.] Zhang, Z., Wu, Y., Zhao, H., Li, Z., Zhang, S., Zhou, X. and Zhou, X., (2020) Semantics-Aware BERT for Language Understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 3405, pp.9628–9635.