SVNIT @ SemEval 2017 Task-6: Learning a Sense of Humor Using Supervised Approach

Rutal Mahajan, Mukesh Zaveri
Computer Engineering Department
S.V. National Institute of Technology, Surat
{rutal.mahajan, mazaveri}@gmail.com

Abstract

This paper describes the system developed for SemEval 2017 task 6: #HashTagWars - Learning a Sense of Humor. Learning to recognize sense of humor is the important task for language understanding applications. Different set of features based on frequency of words, structure of tweets and semantics are used in this system to identify the presence of humor in tweets. Supervised machine learning approaches, Multilayer perceptron and Naïve Bayes are used to classify the tweets in to three levels of sense of humor. For given Hashtag, the system finds the funniest tweet and predicts the amount of funniness of all the other tweets. In official submitted runs, we have achieved 0.506 accuracy using multilayer perceptron in subtask-A and 0.938 distance in subtask-B. Using Naïve bayes in subtask-B, the system achieved 0.949 distance. Apart from official runs, this system have scored 0.751 accuracy in subtask-A using SVM.

1 Introduction

Humor is an integral aspect of human beings that requires self-awareness, spontaneity, linguistic sophistication and empathy. Generating and recognizing humor is not an easy task to be carried out by machines. Generating and understanding humor can be useful in many NLP tasks. (Azizinezhad & Hashemi, 2011) have described the use of humor as the pedagogical tool for language learners, as it helps to keep students interested and motivated. Moreover, recognizing humor is also important in sentiment analysis and opinion mining because it can be useful to get the actual meaning out of figurative sentence.

Research on modeling humor such as (Barbieri & Saggion, 2014)(Raz, 2012)is focused on classifying humor into binary classes as humor and non-humor. In (Reyes, Rosso, & Buscaldi, 2012), humor is modeled by a binary classifier as well as by a multi-class classifier. It classifies different figurative sentences into humor, irony, politics, technology and general sentences. But all these approaches ignore the continuous nature of humor. Hence in task 6 of SemEval 2017 HashTag Wars: Learning a sense of humor (Potash, Romanov, & Rumshisky,2016), humor in tweet should be modeled in its continuous form instead of binary. The participating groups are asked to predict the amount of funniness of the tweet for particular hashtag according to gold labels of tweet. Tweets are labeled with 0, 1, or 2. 0 corresponds to tweet not in top 10. 1 corresponds to tweet in top 10 but not winning tweet and 2 corresponds to winning tweet. There are two subtasks: A) pairwise comparison- a task of predicting which tweet is funnier from given two tweets according to gold labels of tweets. In given pair of tweets, the tweet with higher label is said to be funnier. B) Semi-ranking- a task to predict ranking of tweets from funniest to least funny for given file of tweets for a hashtag.

The remainder of this paper is structured as follows: In section 2, description about overall system architecture is given. It covers pre-processing stage, feature extraction, simple machine learnings approaches for classification and comparator for ranking of tweets. Section 3 describes the results of experiments carried out for subtask A and subtask B by our system followed by conclusion in section 4.

2 System Architecture

This section describes the system architecture submitted for subtask A and B of HashTagWars by the team SVNIT @ SemEval.

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 411–415,
Vancouver, Canada, August 3 - 4, 2017. ©2017 Association for Computational Linguistics

As shown in Figure 1, our system uses simple set of features adapted from (Barbieri & Saggion, 2014) and classifier from weka[1] toolkit.

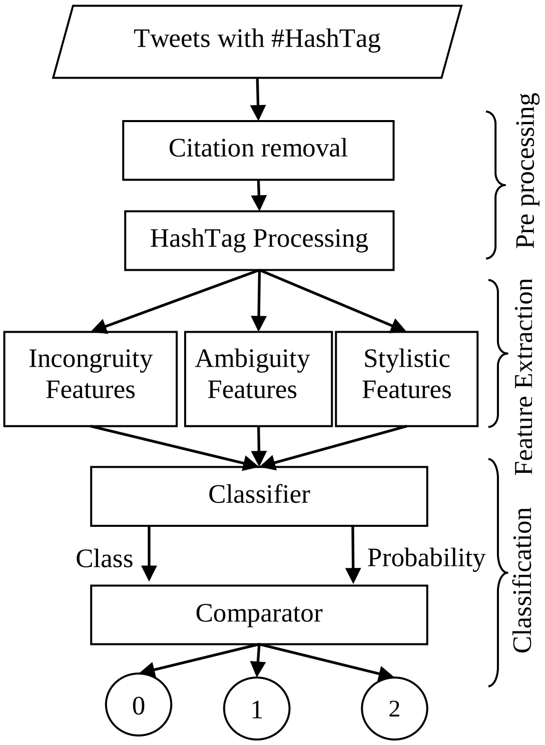

Figure 1: System Architecture of SVNIT@SemEval for HashTag Wars

In the next subsections, different stages of our system for obtaining results of subtask A and B are described.

2.1 Pre-processing

In this stage, the systems takes individual tweets and perform cleaning steps. It removes the references to tweeter username such as @midnight and also processes hash tags. It removes the hash tag from the given tweet and replaces it with corresponding word. E.g. consider tweet in dataset, "See Cats Run. @midnight #CatBooks". Here #CatBooks is replaced with "Cat Books".

2.2 Feature Extraction

After pre-processing of the given tweet, three main set of features are extracted to detect the humor level: 1) Incongruity features; 2) ambiguity features; and 3) stylistic features. Incongruity features checks incongruous or incompatible words in text. E.g. clean *desk* is a sign of cluttered *desk drawer.* Here we use 3 frequency related features of incongruity, and 3 written spoken features. These features are implemented with the help of ANC[2] (American National Corpus). Ambiguity features are important to capture humor in the text as humor is found in two cases: 1) when text has different interpretation and 2) those interpretations are opposed to each other. Here 3 ambiguity related features are used to capture humor in text, which is based on WordNet. Other set of features are Stylistic features, which include 16 features related to structure of the tweet and 8 features related to intensity of adjectives and adverbs. These features are used to detect signatures, unexpectedness and style which is useful for identifying humor in given text (Reyes et al., 2013). Table 1 shows categorization of different features used in this system according to incongruity, ambiguity and stylistic properties captured by them. These groups of features are described below:

Feature name	**No. of features**	**Type**
Frequency related features	3	Incongruity features
Written Spoken Style features	3	
Structure related features	16	Stylistic features
Intensity related features	8	
Synonym related features	4	
Ambiguity related features	3	Ambiguity features

Table 1: Categorization of features based on incongruity, ambiguity and stylistic properties captured by them

Frequency Related Features: Presence of commonly used words and rarest words in tweets are useful to detect unexpectedness and incongruity (Lucariello, 2007; Venour, 2013). We have used ANC frequency corpus for calculating these features. There are three features in this group: 1) Frequency mean is the arithmetic average of frequencies of all words. 2) Rarest word is the frequency value of the rarest word. 3) Frequency gap is the difference between maximum and minimum frequency. For the tweet "A flashlight that doubles as a flesh light. @midnight #BadInventions " , frequency features can be calculated as in Table 2: . For each POS tagged word in tweet written-spoken frequency, written frequency and spoken frequency is calculated respectively as below from

[1] http://www.cs.waikato.ac.nz/ml/weka/

[2] The American National Corpus (http://www.anc.org/) is a massive electronic collection of American English Words (15 million)

ANC corpus. These frequencies are used for the calculation of different frequency features given in Table 2.

A, a: 490433, 406057, 84376
flashlight: 38, 34, 4
that: 98949, 51493, 47456
doubles: 9, 9, 0
as: 107588, 98598, 8990
flesh: 265, 246,19
light: 952,898, 54

Written-Spoken style related features: Informal spoken English is used in many tweets. These features are designed to detect the Incongruity caused by using spoken English in written text or vice versa (Barbieri and Horacio, 2014) (Barbieri and Horacio, 2016). There are three features in this group: 1) Written mean is a mean of frequency values in written ANC corpora. 2) Spoken mean is a mean of frequency values in spoken ANC corpora. 3) Written Spoken gap is the difference between written mean and spoken mean. The example of these feature is given in Table 2.

Feature name	**Value**
Frequency mean	118896.9
Rarest word	0.0
Frequency gap	Max. frequency-Min. frequency = 490433.0
Written Frequency mean	96369.0
Spoken frequency mean	22527.9
Written-Spoken frequency gap	73841.1

Table 3: Example of incongruity features calculation

Structure related features: This group of feature analyzes the structure of given tweet as in (Bertero and Fung, 2016). It uses different structure related features: 1) length is the number of characters in the tweet. 2) Number of words. 3) Word length mean is the mean of word length. 4-7) Number of verbs, nouns, adjectives and adverbs. 8-11) Ratio of above four to total number of words. 6) Number of commas, full stops, ellipsis, exclamation marks and quotation marks.

Intensity related features: We have used Potts (2011) intensity scores to calculate the intensity of adjectives and adverbs. This group of features includes 1) adjective total is the sum of all the adjectives scores. 2) Adjective mean is adjective total divided by number of adjectives. 3) Adjective max is the maximum adjective score. 4) Adjective gap is the difference between adjective max and adjective mean. Similarly, 5) Adverb total 6) adverb mean 7) adverb max and 8) adverb gap is calculated.

Synonyms related features: Some of the humorous tweets convey two messages at the same time (Veale 2004). To identify such a tweet we used this group of features. There are four features in this group. To calculate these features system finds synonyms of all the words using WordNet (Miller 1995) and sorts them according to their ANC frequencies.

This group of features includes 1) synonyms lower mean is the mean of all the synonyms lower. Synonym lower is number of synonyms of word whose frequency is lower than the word's frequency. 2) Synonym lower gap is the difference between word lowest synonym and synonyms lower mean. Word lowest synonym is maximum of synonyms lower. 3) Synonyms greater mean is the mean of all the synonyms greater. Synonym greater is number of synonyms of word whose frequency is greater than the word's frequency. 4) Synonym greater gap is the difference between word greatest synonym and synonyms greater mean. Word greatest synonym is minimum of synonyms greater. For the tweet " Dwarf Cannon. Oh shit, that's actually an AWESOME invention!! #BadInventions @midnight", stylistic feature calculation is given in Table 3.

Feature name	**Value**
Length of tweet	77.0
Number of Words in tweet	13.0
Words Length Mean	4.92307
Number of Verbs	2.0
Number of Nouns	6.0
Number of Adjectives	2.0
Number of Adverbs	1.0
Verb Ratio= Number of Verbs / Total number of words	0.15384
Noun Ratio= Number of Nouns / Total number of words	0.46153
Adjective Ratio= Number of Adjectives / Total number of words	0.15384
Adverb Ratio= Number of Adverbs / Total number of words	0.07692
Number of Commas	1.0
Number of Fullstops	1.0
Number of Ellipsis	0.0
Number of Exclamation	2.0
Number of Quotation	1.0
synoLower Mean	3.18181
synoLower Gap	33.18181
synoGreater Mean	0.0
Syno Greater Gap:0.0	0.0

Table 2: Example of Stylistic features calculation

Ambiguity related features: Three features are used to capture the aspect of Ambiguity as in (Bertero and Fung, 2016). Ambiguity (Bucaria, 2004), the disambiguation of words with multiple meanings (Bekinschtein et al., 2011), is a crucial component of many humor jokes (Miller and Gurevych, 2015; Yang et al., 2015). Features included are 1) synset mean: it is a mean of the number of synsets of each word of the tweet; 2) Max synset: it is a greatest number of synsets that a single word has. 3) Synset gap is a difference between max synset and synset mean.

2.3 Classifiers and Comparator

For classification, we have used simple learning algorithms from Weka, such as implementation of Naïve Bayes classifier and Multilayer perceptron in official submission of subtask A and subtask B. We have also used support vector machine for subtask A and taken results other than official submissions.

In subtask A, we have used Multilayer perceptron (MLP) for pairwise comparison, which initially classifies both the tweets into different classes (0, 1, or 2) then comparator compares the class label of two tweets. Tweet containing higher class label in the pair is considered as funnier tweet. Comparator uses all features and compares given two tweets for the level of humor.

Participating System	**Subtask A (Accuracy)**	**Subtask B (Distance)**
SVNIT@SemEval (unofficial) -SVM	**0.751**	**-**
HumorHawk -run2	**0.675**	-
TakeLab-run2 (unofficial)	0.641	-
HumorHawk -run1	0.637	-
DataStories-run1	0.632	-
Duluth -run2	0.627	**0.872**
TakeLab-run1 (unofficial)	0.597	-
SRHR -run1	0.523	-
SVNIT@SemEval run1- MLP	0.506	0.949
SVNIT@SemEval run2 - NB	-	0.938
TakeLab-run1	0.403	0.908
Duluth-run1	0.397	0.967
TakeLab-run2	0.359	0.944
QUB-run1	0.187	0.924
QUB-run2	-	0.924
#WarTeam	-	1.0

Table 4: result of HashtagWars subtask A and subtask B for all participating systems including unofficial results

In subtask B, using Naïve Bayes classifier and multilayer perceptron tweets are classified into classes among 0, 1 and 2 same as done in subtask A. Tweets with class 1 label are ranked according to their probabilities of class for ranking in funnier to least funny tweet.

3 Experimental Results

In this section, we describe the experiment carried out for the different subtasks and the datasets provided by the organizers. The dataset is composed of 9658 tweets for 86 hashtags roughly collected over seven months of period. Table 4 represents the comparison of result of our system with other systems in subtask A and subtask B as per the results declared on SemEval portal. Scores with bold are best scores of respective system in that subtask.

Our system has scored average in subtask A using Multilayer perceptron classifier with 0.506 accuracy in official submitted runs. In the same subtask our system scored 0.751 accuracy, when evaluated with given evaluation script, which is higher than the highest scoring system. In subtask B, our system have ranked the tweets with 0.938 distance using multilayer perceptron and with 0.949 distance using Naïve Bayes classifier. This edit distance should be as low as possible because it evaluate the system according to how many moves for each tweet need to be occur for placing it at right place.

4 Conclusion

This paper describes the participation of SVNIT at SemEval 2017 task 6 Hashtag wars: learning a sense of humor. We have participated with the system implemented using simplest machine learning algorithms and set of features for humor recognition. Overall our approach using described set of features looks promising but still there is wide room for improvement. We want to improve our machine learning part and set of features by doing error analysis on the achieved results.

References

Azizinezhad Masoud and Hashemi Masoud 2011. Humour: A pedagogical tool for language learners. Procedia - Social and Behavioral Sciences, 30: pages 2093-2098.

http://dx.doi.org/10.1016/j.sbspro.2011.10.407

Barbieri Francesco and Saggion Horacio, 2014. Automatic Detection of Irony and Humor in Twitter. In *Proceedings of the 5th International Conference on Computing Creativity*. Ljubljana, Slovenia, June.

Barbieri Francesco and Saggion Horacio, 2016. Irony Detection in Twitter: The Role of Affective Content. *ACM Transactions on Internet Technology (TOIT)*, 16(3), July, pages 19:1—19:24. http://doi.acm.org/10.1145/2930663

Dario Bertero and Pascale Fung, 2016. Predicting humor response in dialogues from TV sitcoms. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai March 20-25, pages 5780-5784. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7472785&isnumber=7471614

Diyi Yang, Alon Lavie, Chris Dyer and Eduard Hovy, 2015. Humor Recognition and Humor Anchor Extraction, In *Proceedings of 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, September 17-20, Association for Computational Linguistics, pages 2367-2376.http://aclweb.org/anthology/D/D15/D15-1284.pdf

Peter Potash, Alexey Romanov and Anna Rumshisky, 2017. SemEval-2017 Task 6: #HashTagWars: Learning Sense of Humor. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval 2017)*. August. Association for Computational Linguistics.

Raz Yishay. 2012. Automatic Humor Classification on Twitter. *Proceedings of the NAACL HLT 2012 Student Research Workshop*, Montreal, Canada. Association for Computational Linguistics, pages 66–70.http://www.aclweb.org/anthology/N12-2012

Reyes Antonio, Rosso Paolo and Buscaldi Davide. 2012. From humor recognition to irony detection: The figurative language of social media." *Data and Knowledge Engineering*, 74, April, pages 1–12. https://doi.org/10.1016/j.datak.2012.02.005

Reyes Antonio, Rosso Paolo and Veale Tony, 2013. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation*, Springer-Verlag New York, 47(1): pages 239-268. http://dx.doi.org/10.1007/s10579-012-9196-x

Veale Tony, 2004. The challenge of creative information retrieval. In *Proceedings of Computational Linguistics and Intelligent Text Processing:5th International Conference, CICLing*. February 15-21, Springer Berlin Heidelberg, pages 457–467. http://dx.doi.org/10.1007/978-3-540-24630-5_56

Venour Chris. 2013. A computational model of lexical incongruity in humorous text. Ph.D. Dissertation, University of Aberdeen.