Countering an Anti-Natural Language Processing Mechanism in the Computer-Mediated Communication of "Trusted" Cyberspace Operations

Bi-Normal Separation Feature Scaling for Informing a Modified Association Matrix for Enhanced Event Correlation

Steve Chan
Decision Engineering Analysis Laboratory
San Diego, California U.S.A.
email: schan@dengineering.org

***Abstract*—There are bypass mechanisms to Natural Language Processing capabilities, such as the usage of irony, sarcasm, and satire, particularly as pertains to Computer-Mediated Communications. The problem then is that of the gradations between irony, sarcasm, and satire. Irony is used to convey, usually, the opposite meaning of the actual things said, but its purpose is not necessarily intended to hurt the target. The purpose of sarcasm, unlike irony, is to hurt the target. Satire might utilize irony, exaggeration, ridicule, and/or humor to expose and criticize shortcomings and/or vices of the target. The detection of these usages is an intriguing challenge. For example, sarcasm detection is difficult as there are several gradations; sarcasm might be comprised of real sarcasm, semi-irony, or friendly sarcasm. Determining the cognitive context, which triggered the original manifestation remains a bridge to be solidified. Also, sarcasm detection often exceeds even the concept of context, as it can be distorted by either the sender and/or receiver. This remains a herculean challenge in the domain, as others remain focused on first-order metarepresentations (e.g., analogies), while the challenges of second-order metarepresentations are more sparsely addressed. This paper presents a possible framework to address the problem by utilizing Bi-Normal Separation Feature Scaling for informing a Modified Association Matrix as contrasted to a framework utilizing Inverse Document Frequency and a prototypical Association Matrix. It is posited that the former will exhibit faster convergence and accuracy for enhanced detection of irony, sarcasm, as well as satire, and preliminary results seem to indicate this. The main output of the paper is a potential solution stack that directly contends with the second-order metarepresentation issue.**

Keywords-Satire; Natural Language Processing; Deep Learning; Dimensionality Reduction; Bi-Normal Separation Feature Scaling; Modified Association Matrix.

I. INTRODUCTION

Computer-Mediated Communication (CMC) has become prevalent (e.g., in-game text-based chat) in Massively Multiplayer Online Games (MMOGs) (e.g., World of Warcraft) and digital media entertainment services (e.g., Playstation Network). This trend is increasing as various open-source chat Software Development Kits and Application Programming Interfaces (e.g., PubNub ChatEngine) become available for the developers in a growing gaming industry. The vulnerabilities presented by CMC are discussed within literature. In an academic sense, if Natural Language Processing (NLP) were applied to this type of chat traffic, the analysis would be more challenging due to the variety of newly coined jargon words, etc. that continually emerge in this domain. However, the use of elevated language conjoined with satire constitutes an even greater challenge and a recipe for an anti-NLP (making a comparison to Anti-Face) mechanism that potentially poses a threat that impacts "trusted" cyberspace.

The remainder of this paper is organized as follows: Section II provides a primer by discussing some advantages of Inverse Document Frequency (IDF) over the simplicity of the Poisson distribution. Subsequently, Section III provides additional background information by discussing some advantages of neural embeddings (a.k.a. word embeddings) over n-grams. Then, Section IV delves into the complexities of the computational processing associated with figurative language as compared to literal language. Section V discusses some advantages of transfer learning (with bi-normal separation feature scaling) over deep learning in addressing the challenges of figurative language as well as posited improvements over IDF. Section VI discusses optimizing the [deep] transfer learning convolutional neural network inference engine, which was discussed in Section V. Section VII further discusses optimizing the inference engine with a modified association matrix. Section VIII posits a framework for the enhanced experimental inference engine, particularly as pertains to irony, sarcasm, and satire detection. Section IX presents the experimental results from the experimental inference engine solution stack, which incorporates the elements discussed in Section V, Section VI, and Section VII. Finally, the paper reviews and emphasizes key points within Section X, the conclusion.

II. FROM POISSON TO INVERSE DOCUMENT FREQUENCY

NLP pertains to the interactions between computers and human languages. In the operationalization of primordial NLP, word frequency is often practiced, as the following logic is utilized: "Low frequency words tend to be rich in

Copyright (c) IARIA, 2018. ISBN: 978-1-61208-683-5

content, and vice versa" [3]. This logic focuses upon "rare words," and there is an implicit assumption that words (e.g., n-grams) are distributed by a single parameter distribution, such as a Poisson process or a binomial. However, these distributions do not fit data very well; in fact, the Poisson distribution predicts that "lightning is unlikely to strike twice (or half a dozen times) in a single document" [4]. According to this logic, there should not be an expectation of seeing two or more instances of a "rare word" in a single document (unless there is some sort of hidden dependency that goes beyond the Poisson [4]; generally speaking, the utilization of Poisson for modeling the distribution of words [e.g., n-grams] fails to fit the data except in the case wherein there are almost no interesting dependencies). Yet, dependencies are indeed prevalent [8], and many NLP applications endeavor to discriminate documents on the basis of certain hidden variables, such as topic, author, genre, style, and the like [9]. The more that a keyword (e.g., n-gram) deviates from Poisson, the stronger the dependence on hidden variables, and the more useful, potentially, the n-gram is for discriminating documents on the basis of these hidden dependencies.

In the modern age of search engine optimization (which will include, among other techniques, keyword density), the likelihood of seeing a "rare word" is actually quite high. Hence, the employing of word frequency (a.k.a. raw frequency) has an inherent deficiency for the task-at-hand, as all terms are arbitrarily given equal weighting as pertains to assessing relevancy for a query. For example, a collection of documents discussing the "game" industry is likely to have the term "game" in almost every document. To mitigate this particular effect of certain "rare words" (a.k.a. "rare terms"), which occur too frequently within the collection (so as to be meaningful for relevance determination), an IDF mechanism is often utilized. Indeed, much better fits are obtained by introducing a second parameter, such as IDF, which is defined as $-\log 2df_w/D$, where D is the number of documents in the collection and df_w is the document frequency (i.e., the number of documents, which contain w); observationally speaking (e.g., as contrasted to Poisson), the IDF for a "rare term" is high, whereas the IDF of a "frequent term" is likely to be low [10]. This is consistent with the current notion that words with larger IDF tend to have more inherent content, and a good "[rare] word" should be located farther from the chance of Poisson [11].

III. FROM N-GRAMS TO NEURAL EMBEDDINGS

This paper endeavors to address a key NLP problem known as *sarcasm detection* by utilizing a combination of models based upon, among others, specifically engineered Convolutional Neural Networks (CNNs). The automated detection for an expression of sarcasm is non-trivial and, in many cases, involves a reversal of the polarity of a sentence. By way of example, "I love working eighty hours a week to be this poor" is such an expression of sarcasm. Another commonly cited example is "I love the pain of breakup" [12]. In both cases, it is difficult to extract the requisite crystalline aspects needed to determine the presence of sarcasm within the sentence. The example, "I love working eighty hours a week" provides aspects of an expressed sentiment (in this case, that of a positive nature), and "to be this poor" describes a contradicting sentiment (that of a negative nature). Hence, we dissect these amalgams.

In the realm of language, a simile compares one thing to another (similes are more likely to utilize the words "like" and "as," which a metaphor does not utilize. Verbal irony refers to the use of vocabulary to describe something in a way that is other than what it seems; indeed, verbal irony can consist of "ironic similes," which are comparisons between two items that are not alike at all [e.g., "fire and ice"]). Hao and Veale conducted various experiments over a corpus of "ironic similes" in which the authors found that most of the examined ironic comparisons utilize a precursor positive sentiment to impart a negative view (~70%) [13]. They found that sarcasm is very topic-dependent and highly contextual. Thus, sentiment and other contextual clues are vital for *sarcasm detection*, and this is at the core of anaphora resolution.

Certain features (e.g., n-grams), although somewhat useful for *sarcasm detection*, produce very sparse (often too sparse) feature vector representations or sparse vectors, and this had led to a surge of work centered upon representing words as dense feature vector representations or dense vectors. These representations, referred to as "word embeddings" or "neural embeddings," have been shown to perform well for a variety of NLP tasks. For example, in the often utilized *word2vec*, a distributed representation of a word is used in the form of a vector with many dimensions. Each word is represented by a distribution of weights across those elements. Hence, instead of a one-to-one mapping between an element in the vector and a word, the representation of a word is spread across all of the elements within the vector, and each element in the vector contributes to the definition of many words. The multi-dimensional vector comes to represent, abstractly, the "meaning" of a word. In essence, there is a word-context matrix M for which each row i corresponds to a word, each column j corresponds to a context for which the word has appeared, and each matrix entry M_{ij} corresponds to some association measure between the word and the context. Words are then represented as rows in M or in a dimensionality-reduced matrix based upon M. By examining a large corpus of text, it is possible to ascertain word vectors that are able to capture relationships between words in a surprisingly expressive way. Along this vein, these vectors can be utilized as inputs to a deep learning CNN. It is found that these CNN-learned word representations well capture meaningful syntactic and semantic regularities [14]. Along this vein, many NLP tasks benefit from word representations that do not treat individual words as unique symbols, but instead reflect similarities and dissimilarities between them.

The common paradigm for deriving such representations is based upon the distributional hypothesis of Harris [15], which asserts that words in similar contexts have similar meanings. Dingemanse et al. claim that universal words (e.g., "huh") occur in a large sample of unrelated languages and have similar contexts [16]; consequently, affirmation of context (e.g., same contexts in different languages) can be useful.

Specifically, the regularities observed are translated into constant vector offsets between pairs of words sharing a particular relationship. In fact, these vectors are very good at answering analogy questions of the form a is to b as c is to ?. However, CNNs are time-consuming to train, so other models are often utilized that might not be able to represent the data as precisely as CNNs but can often be trained efficiently on much more data. By way of example, the previously discussed *word2vec* is similar to an autoencoder. However, rather than training against the input words, via reconstruction, as a restricted Boltzmann machine does, *word2vec* trains words against other neighboring words in the input corpus, via two exemplar models: (1) Continuous Bag-of-Words (CBOW), using context to predict a target word, and (2) Skip-Gram, using a word to predict a target context. The latter model is often utilized, as it produces more accurate results on large datasets.

IV. From Literal to Figurative Language

In NLP, Word Sense Disambiguation (WSD) is the challenge of determining which "sense" (i.e., meaning) of a word is activated by the use of the word in a particular context. Literal language means exactly what it says. In contrast, figurative language represents one of the most difficult tasks for NLP. Several types of figurative language include personification, hyperbole, idioms, onomatopoeia, simile, and metaphor. Lakoff and Johnson assert that metaphor is a method for transferring knowledge from a concrete domain to an abstract domain (a first-order metarepresentation), and they posit that the degree of abstractness in a word's context is correlated with the likelihood that the word is used metaphorically [17]. Consider the following sentences: (L) He shot down my plane, and (M) He shot down my argument. The literal sense of "shot down" in L invokes knowledge from the domain of war [17]. The metaphorical usage of "shot down" in M transfers knowledge from the concrete domain of war to the abstract domain of debate [17]. Danesi contends that metaphor transfers associations from the source domain to the target domain [18]. Accordingly, the metaphorical usage of "shot down" in M carries associations that are not conveyed by L. To contend with the abstractness, various approaches are utilized for textual inference; one such approach is that of Recognizing Textual Entailment (RTE) [19]. To posit correct inferences, as pertains to RTE, systems must be able to distinguish between the literal and metaphorical senses of a word, and the degree of abstractness of words is one approach. For instance, the "plane" in L is rated with a lower number (i.e., relatively concrete), whereas "argument" in M is rated with a higher number (i.e., relatively abstract), which suggests that the verb "shot down" is used literally in L, whereas it is used metaphorically in M [17]. Turney and Littman rated words according to their semantic orientation, such as denotative (i.e., literal) or connotative (i.e., non-literal, metaphorical); by way of example, "deep mud" is labeled as denotative, and "deep gratitude" is labeled as connotative.

Unlike literal language, figurative language utilizes linguistic devices (e.g., simile, metaphor) to communicate indirect meanings (e.g., sarcasm) which, usually, are not readily interpreted by simply decoding syntactic or semantic information. Indeed, figurative language reflects patterns of thought that are not only challenging in their linguistic representations, but also for the involved requisite computational processing; figurative language processing can involve a variety of processes, such as sentiment analysis or opinion mining. Katz et al. posit that irony tends to be more difficult to comprehend than metaphor because irony requires the ability to recognize, at the very least, a second-order metarepresentation [20]. This same notation applies to sarcasm and satire; accordingly, irony, sarcasm, and satire constitute second-order metarepresentations.

V. From Prototypical Deep Learning to Transfer Learning with Bi-Normal Separation Feature Scaling

To address the aforementioned challenges, deep learning is often utilized. Prototypical deep learning can be broken down into two parts: training and inference. When the deep learning CNN has been well trained on what to detect, the inference engine proceeds to make inferences or predictions based upon the input data. In general, deep learning requires a prodigious amount of data for training. Unfortunately, collecting this data for niche areas, where data is typically sparse, is challenging. One approach towards resolving this dilemma is known as Transfer Learning, wherein the model becomes trained on other datasets (i.e., pre-training), and weights for each layer are assigned in a "rough-tuned" fashion, iteratively. Hence, instead of initializing the weights for each layer randomly, as is typically done for models being trained from scratch, the learned weights for each layer of the pre-trained model are "fine-tuned" during the training on the sparse data. Theoretically, this TL paradigm has a better chance of converging much more quickly, and it is typically achieved, via the following pathways:

A. Continuous Back Propagation

The involved pre-trained model can be further "fine-tuned" by continuing the back propagation and updating the weights of all the layers. Alternatively, only certain layers may be "fine-tuned." Comprehensively, the "fine-tuning" can start at the highest-level layer and progress towards the lowest-level layer with a continuous assessment of the

Copyright (c) IARIA, 2018. ISBN: 978-1-61208-683-5

performance and determinations made accordingly along the way in terms of tuning.

B. Hybridizing CNN with Support Vector Machine

The involved pre-trained model can also serve as a feature extractor for the data. These features can then be fed into a linear classifier, such as a Support Vector Machine (SVM). This hybridized approach is ideal if the dataset is particularly sparse and "fine-tuning" the model is likely to result in over-fitting.

C. Bi-Normal Separation Feature Scaling

The described pre-trained model can, ideally, infer — by way of example — what decision is likely to be made next. Ideally, the pre-trained model can make robust inferences from new data based upon its prior training. In the realm of NLP, wherein the numerical feature value for a given word/term is often represented by its Term Frequency (TF) (within the given text) multiplied by its IDF (within the entire corpus), the "TF·IDF" combinatorial has become a prevalent representation [21]. However, IDF is oblivious to the training class labels and will, as a consequence, scale some features inappropriately. In contrast, Bi-Normal Separation (BNS) feature scaling has been shown to outperform other feature representation schemes for a wide range of text classification tasks. The superiority of BNS is especially pronounced for collections with a low proportion of positive class instances. With BNS, features are allocated a weight according to $|F^{-1}(tpr) - F^{-1}(fpr)|$, where F^{-1} is the Inverse Normal Cumulative Distribution Function (INCDF), tpr is the true positive rate (P(feature|positive class)), and fpr is the false positive rate (P(feature|negative class)). BNS produces the highest weights for features that are strongly correlated with either the negative or positive class. Features that occur fairly evenly across the training instances are given the lowest weight. Furthermore, BNS scaling has yielded better performance even without feature selection, potentially obviating the need for such [21]. This accelerates the performance of the inference engine.

VI. Optimizing the Transfer Learning CNN

There are two main approaches to modifying the TL CNN for reducing latency, particularly in the case of applications operating across other networks. The first approach is that of eliminating layers of the CNN that are not activated after training. The second approach is that of combining various layers of the CNN into a single computational step. These approaches should result in a similar accuracy of prediction, but simplified, compressed, and optimized for runtime performance (some compare this to the case of optimizing [i.e., compressing] an image for the WWW; ideally, the differences between the uncompressed and compressed image will be indistinguishable to the human eye) [22]. In essence, this further accelerates the performance of the inference engine.

VII. Modified Association Matrix

A final accelerant for the inference engine comes by way of a Modified Association Matrix (MAM). A typical Association Matrix (AM) is a 2-dimensional matrix, wherein each cell c_{ij} represents the correlation factor between the terms in the query and the terms in the documents. This matrix is used to reformulate an original query to improve its performance [23]. Each correlation factor, denoted as c_{ij} is calculated in accordance with (1):

$$c_{ij} = \sum_{d_k \in D} f_{ik} \times f_{jk} \tag{1}$$

where c_{ij} is the correlation factor between term i and term j, and f_{ik} is the frequency of term i in document k. Additionally, these correlation values are used to calculate the normalized association matrix in accordance with (2):

$$s_{ij} = c_{ij}/(c_{ii} + c_{jj} - c_{ij}) \tag{2}$$

where s_{ij} denotes the normalized association score, and c_{ij} represents the correlation factor between term i and term j. A higher normalized association score implies a higher degree of correspondence with the original query [23]. Words with the highest association scores are selected to be added back into the original query, and this new query (instead of the original query) is utilized to calculate cosine similarity. This new query, theoretically, should have a similar profile to the intent of the formulation of the query, and this will be reflected, via cosine similarity [24].

With regards to intent, Hancock had examined differences in verbal irony usage, via face-to-face and CMC, and found that irony (specifically, sarcasm) was more common in CMC settings and was primarily signaled through punctuation [25]. Reyes & Rosso utilized a corpus of review comments regarding products on Amazon.com, and they utilized six factors for their model: N-Grams (NG) (i.e., recurrent word combinations), Part-of-Speech (POS) N-Grams (POSNG) (i.e., recurrent POS combinations), words with semantic characteristics of sexuality or relationships (using values from WordNet), Positive and Negative Values (PNV) of words (using values from the Macquarie Semantic Orientation Lexicon [MSOL]), Pleasantness Value (PV) of words (using values from Whissel's Dictionary of Affect in Language [WDAL]), and Affective words Demonstrating Subjectivity (ADS) (using values from WordNet) [25]. The model utilized for this paper had some deletions and incorporated some modifications to the Reyes & Rosso model. For example, the MSOL was conjoined with the Yelp Restaurant Sentiment Lexicon (YRSL) and the Amazon Laptop Sentiment Lexicon (ALSL). The WDAL was complemented by the National Research Council (NRC) [Canada] Hashtag Emotion Lexicon (HEL) and the NRC Word-Emotion Association Lexicon (WEAL).

By applying specific scaling methods on an association matrix, unigrams were placed into a higher dimensional space, such that the unigrams with similar associative patterns were placed in similar regions of the dimensional space. This resultant space is referred to as the Word Association Space (WAS). The number of dimensions will vary depending on how much of the information of the free association database is compressed, but intermediate values between 200 and 500 are to be expected [26]. Typically, the dimensionality of the WAS equates to the number of features for the unigrams. Of note, with too few dimensions, the similarity structure of the resulting vectors does not capture enough granularity of the original associative structure within the free association database. With too many dimensions (e.g., the number of dimensions approaches the number of features), the information is not compressed enough, and the similarity structure of the vectors does not capture enough of the indirect relationships regarding the associations between the involved unigrams. Overall, this modified association matrix results in an enhancement of the first-order metarepresentation. In addition, the associations between the first-order metarepresentations are enhanced. Accordingly, the reversals of the polarity of sentences are better detected by the MSOL, YRSL, and ALSL-trained CNN. These lexical databases provide a more robust corpus for the sentiment of various words and are further buttressed by WDAL, HEL, and WEAL-training. By way of example, “I love working eighty hours a week to be this poor” can now be identified as an amalgam of two first-order representations that involve a reversal of polarity. Given this reversal of polarity identification, the involved first-order metarepresentations can be tagged and associated with sarcasm. This then segues to an improvement of the second-order metarepresentation for *sarcasm detection*.

VIII. Posited Framework for an Enhanced Inference System, Particularly for Irony, Sarcasm, and Satire Detection

A prototypical Deep Learning Engine (Training Engine and Inference Engine) with the specified components for the experiment is as delineated in Figure 1. Several exemplar layers (NG, POSNG, PNV, PV, and ADS) are provided for the Training Engine. These same exemplar layers are utilized for both the Forward Propagation “Rough-Tuning” for the Training Model as well as the Continuous Back Propagation “Fine-Tuning” for the Pre-Trained Model. As discussed previously in Sections II through VII, BNS and MAM may be leveraged as accelerants for the inference engine. When combined with a specifically chosen datasets to assist in the pre-training, the Transfer Learning is enhanced. By way of explanation, the Untrained Model eventually becomes a “Rough-Tuned” Trained Model (upon ingestion of the initial Training Dataset and Forward Propagation). Further “Rough Tuning” can be achieved by training specific layers, such as PNV and PV (e.g., via MSOL and WDAL, respectively). Eventually, the Trained Model becomes a Pre-Trained Model, and “Fine-Tuning” can be achieved by Continuous Back Propagation and optimizing at certain training layers, such as PNV (e.g., via YRSL and ALSL), and PV (e.g., via HEL and WEAL). The Pre-Trained Model is then further optimized when the New Dataset is Ingested. To avoid over-fitting, the Pre-Trained Model of the CNN can also serve as a feature extractor for which the features can be fed into an SVM. Collectively, the hitherto described constituent components constitute the posited framework for an enhanced TL CNN inference engine for *sarcasm detection.*

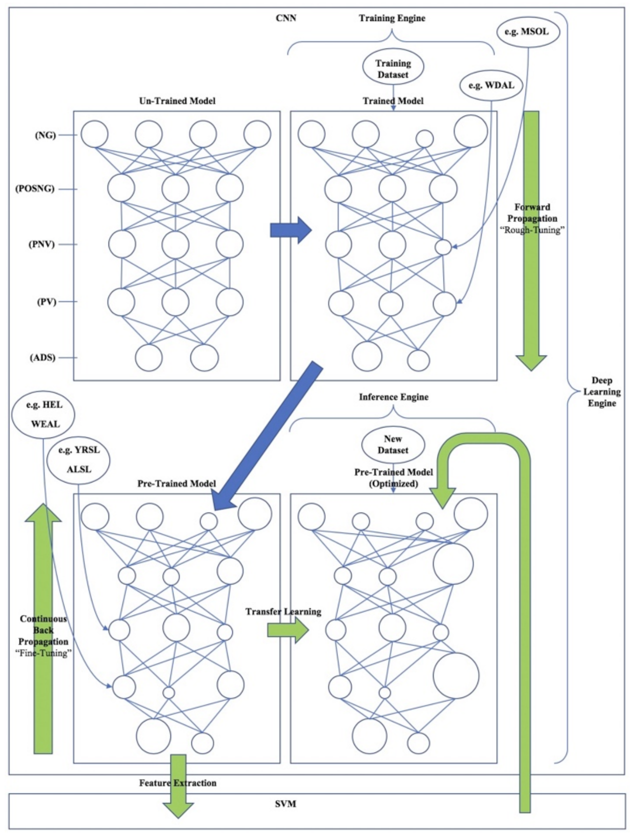

Figure 1. Posited Experimental Framework For an Enhanced Inference System, Particularly for Irony, Sarcasm, and Satire Detection.

Overall, the posited experimental framework comprised of a prototypical Deep Learning Engine (Training Engine and Inference Engine) with various enhanced layers (NG, POSNG, PNV, PV, and ADS) for the Training Engine seems to be quite useful for the Forward Propagation “Rough-Tuning” process for the Training Model as well as the Continuous Back Propagation “Fine-Tuning” process for the Pre-Trained Model.

IX. Preliminary results from the experimental inference engine solution stacks

The hitherto described solution stacks are as follows: (1) Solution Stack #1: IDF & AM, and (2) Solution Stack #2: BNS & MAM. This is shown in Figure 2 below as Phase 1 of the experiment for Solution Stack #1 and Phase 2 of the experiment for Solution Stack #2. The preliminary results are also shown in Figure 2 and indicate a 9% performance edge by Solution Stack #2 over Solution Stack #1 when benchmarked against the Self-Annotated Reddit Corpus (SARC), a corpus of 1.3 million sarcastic remarks. SARC was chosen as it had both sarcastic and non-sarcastic comments, thereby allowing for learning in both balanced and unbalanced label regimes [27]. However, there are also deficiencies with SARC. By way of explanation, Reddit users have adopted a common method for sarcasm annotation consisting of adding the marker "/s" to the end of sarcastic statements (this originates from the HTML usage <sarcasm>...</sarcasm>). As with Twitter hashtags, using the markers "/s" as indicators of sarcasm "is noisy, for many users do not use the marker, do not know about it, or only use it where sarcastic intent is not otherwise obvious" [27]. The experiment also has not yet treated the case of false positives and false negatives. Further investigation is needed, as these are preliminary results only, and only one "New Dataset" was utilized for testing.

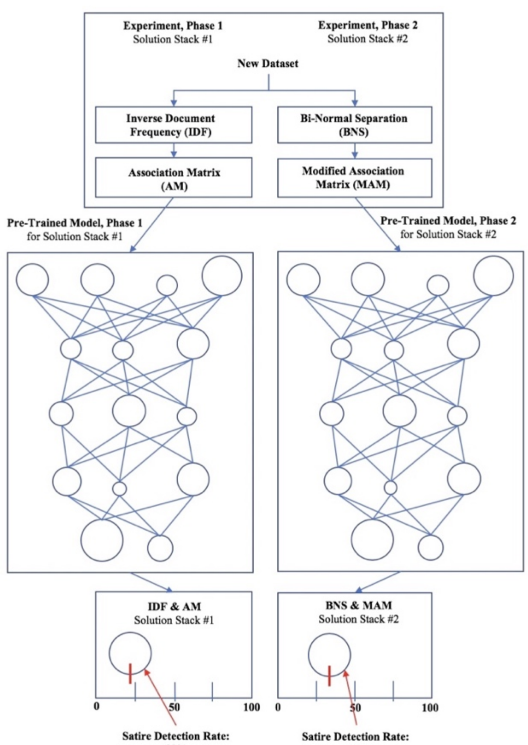

Figure 2. Solution Stack #1 (IDF & AM) versus Solution Stack #2 (BNS & MAM): Solution Stack #2 demonstrates a 9% performance edge over Solution Stack #1 (for a single test case).

Overall, Solution Stack #2 from Phase 2 of the experiment seems to have an advantage over Solution Stack #1 from Phase 1 of the experiment when benchmarked against the SARC. However, there is much more work to be done regarding false positives and false negatives.

X. Conclusion

This paper presents the benchmarked performance results of a posited framework for an enhanced inference system. The premise for devising such a system was predicated on the problems with *sarcasm detection* (i.e., detecting for a second-order metarepresentation) within the NLP arena. The described work utilized SARC as well as a variety of datasets for a Pre-Trained Model. The preliminary results of an approximately 9% performance edge by Solution Stack #2 (BNS & MAM) over Solution Stack #1 (IDF & AM) seem promising, but only one test was performed. Future work necessitates a further investigation with a much more robust performance metric and benchmarking paradigm, as well as the potential involvement of other useful viable datasets for fine-tuning of the CNN Pre-Trained Model. An updated literature review will be performed for updated techniques and methodologies.

Acknowledgment

The author would like to thank the Decision Engineering Analysis Laboratory (DEAL) for its encouragement and motivation throughout the process of pursuing and completing this research. Without their initial and continuing assistance, as well as the ideas, feedback, suggestions, guidance, resources, and contacts made available through that support, much of this research would have been delayed. The author would also like to thank VT & IE^{2}SPOMTF. This is part of a paper series on enhanced event correlation. The author would like to also thank USG leadership (e.g., the CAG). The author would further like to thank the International Academy, Research, and Industry Association (IARIA) for the constant motivation to excel as well as the opportunity to serve as a contributing IARIA Fellow within the cyber and data analytics domains, particularly in the area of mission-critical systems.

References

[1] New Jersey Fusion Center, "Bronx Bloods Members Communicating Through PlayStation Network (PSN)," Federal Bureau of Investigation New York, pp. 1-3, 28 July 2011.

[2] M. Ruskin, "Playing in the Dark: How Online Games Provide Shelter for Criminal Organizations in the Surveillance Age," Arizona Journal of International and Comparative Law, vol. 31, pp. 875-906, 2014.

[3] K. Church and W. Gale, "Inverse Document Frequency (IDF): A Measure of Deviations from Poisson," vol. 11, pp. 283-295, 1999.

[4] K. Church and W. Gale, "Poisson Mixtures," vol. 1, pp. 163-190, June 1995.

[5] D. Bahdanau et al., "Learning to Compute Word Embeddings on the Fly," ArXiv, pp. 1-12, 7 March 2018.

[6] M. Luong, I. Sutskever, Q. Le, O. Vinyals, and W. Zaremba, "Addressing the Rare Word Problem in Neural Machine Translation," Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL) and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, vol. 1, pp. 11-19, 2015.

[7] M. Evans, "Probability and Statistics: The Science of Uncertainty," Content Technologies, Inc., 2013.

[8] S. Khudanpur and J. Wu, "A Maximum Entropy Language Model Integrating N-grams and Topic Dependencies for Conversational Speech Recognition," IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 553-556, 1999.

[9] S. Armstrong et al., "Natural Language Processing Using Very Large Corpora," Springer-Science+Business Media, B.V., 1999.

[10] C. Manning, H. Schutze, and P. Raghavan, "Introduction to Information Retrieval," Cambridge University Press, 7 July 2008.

[11] C. Manning and H. Schutze, "Foundations of Statistical Natural Language Processing," MIT Press, 1999.

[12] S. Poria, E. Cambria, D. Hazrika, and P. Vij, "A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks," 26th International Conference on Computational Linguistics (COLING 2016), pp. 1601-1612, 2016.

[13] A. Perez, "Linguistic-based Patterns for Figurative Language Processing: The Case of Human Recognition and Irony Detection," Universitat Politecnica De Valencia, pp. 107-109, July 2012.

[14] T. Mikolov, W. Yih, and G. Zweig, "Linguistic Regularities in Continuous Space Word Representations," Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies (HLT), pp. 746-751, June 2013.

[15] M. Sahlgren, "The Distributional Hypothesis," Italian Journal of Linguistics, vol. 20, pp. 1-18, 2008.

[16] M. Hayashi, G. Raymond, and J. Signell, "Conversational Repair and Human Understanding: Huh? What? – A First Survey in 21 Languages," Cambridge University Press, pp. 343-380, 2013.

[17] P. Turney, Y. Neuman, D. Assaf, and Y. Cohen, "Literal and Metaphorical Sense Identification through Concrete and Abstract Context," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 680-690, 2011.

[18] S. Mohammad, E. Shutova, and P. Turney, "Metaphor as a Medium for Emotion: An Empirical Study," Proceedings of the Joint Conference on Lexical and Computational Semantics, pp. 23-33, 2016.

[19] I. Dagan, D. Roth, M. Sammon, and Fabio Zanzotto, "Recognizing Textual Entailment: Models and Applications," Morgan & Claypool, pp. 157-159, 2013.

[20] L. Bobrova and J. Lantolf, "Metaphor and Pedagogy," CALPER Working Paper Series, Center for Advanced Language Proficiency Education and Research, pp. 1-34, 2012.

[21] G. Forman, "BNS Feature Scaling: An Improved Representation over TF•IDF for SVN Text Classification," Proceedings of the 17th Association for Computing Machinery (ACM) Conference on Information and Knowledge Management, pp. 263-270, 2008.

[22] M. Copeland, "What's the Difference Between Deep Learning Training and Inference?" Nvidia, pp. 1-2, 22 August 2016.

[23] A. Boutari, C. Carpineto, R. Nicolussi, "Evaluating Term Concept Association Measures for Short Text Expansion Two Case Studies of Classification and Clustering," Proceedings on Extending Database Technology (EDBT), pp. 163-174, 2010.

[24] J. Chartee, E. Cankaya, and S. Phithakkitnukoon, "Query Expansion using Association Matrix for Improved Information Retrieval Performance," Artificial Intelligence and Hybrid Systems, iConcept Press, pp. 1-6, 2013.

[25] S. Skalicky and S. Crossley, "A Statistical Analysis of Satirical Amazon.com Product Review," European Journal of Humour Research, vol. 2, pp. 66-85, 31 July 2018.

[26] M. Steyvers, R. Shiffrin, D. Nelson, "Word Association Spaces for Predicting Semantic Similarity Effects in Episodic Memory," Decade of Behavior: Experimental Cognitive Psychology and its Applications, pp. 237-249, 2014.

[27] M. Khodak, N. Saunshi, and K. Vodrahalli, "A Large Self-Annotated Corpus for Sarcasm," The International Conference on Language Resources and Evaluation (LREC), pp. 1-6, 22 March 2018.