Investigation on the application of MSASB for speaker recognition.

Vennela Miryala[1], Sai Sirisha Rallabandi[1], Sivanand Achanta[1], Sai Krishna Rallabandi[1] and Suryakanth V gangashetty[1]

vennelamiryala@gmail.com, {sirisha.rallabandi, sivanand.a, saikrishna.r}@research.iiit.ac.in, svg@iiit.ac.in

[1]International Institute of Information Technology, Hyderabad

ABSTRACT

In this paper we are investigating the use of the maximum spectral amplitude in sub-bands(MSASB) in the Speaker Recognition task.So far,most of the research has been done in this aspect using MFCCs,LPCCs,I vector.In this paper we are able to show that MSASBs can differentiate the speakers. Also experiments are also done to observe the compressibility of them.

General Terms

Maximum Spectral Amplitude in Sub-bands,Mel-Frequency Cepstral Coefficients,Deep Neural Networks

1. INTRODUCTION

Automatic humour recognition is a small but growing area in the field of Information Retrieval.The complexity of the problem lies in identifying the constituting element of the joke that delivers humour. Reyes et al. [**?**] prove that features like ambiguity (morphosyntactic and semantic), perplexity and sentence complexity can be used to distinguish between humorous and non-humorous one-liners. Their results show that humorous one-liners show a higher degree of ambiguity for all these features. Strapparava et al.[**?**] study automatic classification of humour of one-liners using stylistic features, content-based features or both in a learning framework. Stuart et al. [**?**] proposes a template-match based recognition system using Ontological System Technology and constructionist approaches. Zhang et al. [**?**] propose a systematic method to derive the correlation between humor-related aspects and features from humor-related theories.

Although each of these techniques work quite well on a limited domain, they rely heavily on hand crafted features. Recently, vector space models of word embeddings have been very successful at learning both the lexical as well as semantic information and are applied to a variety of useful natural language processing applications such as search query expansions [**?**], quick extraction for IR [**?**], automatic annotation [**?**]. In all these models, the meaning of a word is encoded as a vector computed from the co-occurence statistics of a word and its neighboring words.They are shown to capture the syntax [**?**],[**?**],[**?**]. These models can be extended to automatic humor recogniton.

In this paper, we propose a framework that uses word embeddings learnt from a recursive neural network for automatic recognition of humor from social media data. In Section 2, we briefly describe the word embedding scheme and in Section 3 we describe the model design, followed by experimental results and conclusion.

2. EMBEDDING

Embedding refers to the process where words from the vocabulary (and possibly phrases thereof) are mapped to vectors of real numbers in a low dimensional space, relative to the vocabulary size ("continuous space").

2.1 Word Vector Representations

The following are the various publicly available pretrained word vectors which can be used to test the applicability of any task which uses embedding. These representations exhibit a balanced mix between large and small amounts of unlabeled text as well as between the neural and spectral methods of training word vectors.

2.1.1 Glove Vectors

Global vectors for word representations [**?**] are trained on aggregated global word-word co-occurrence statistics. These vectors were trained on 6 billion words from Wikipedia and English Gigaword, and are of length 300.

2.1.2 SkipGram Vectors

Mikolov et al. [**?**] proposed the Word2Vec tool. In this model, Huffman code of each word is used as an input to a log-linear classifier with a continuous projection layer and words within a given context window are predicted. The available vectors are trained on 100 billion words of Google news dataset and are of length 300.

2.1.3 Global Context Vectors

These vectors are learned using a recursive neural network [**?**] that incorporates both local and global (document-level) context features [**?**]. These vectors were trained on the first 1 billion words of English Wikipedia and are of length 50.

2.2 Semantic Lexicons

A lot of optimization techniques have been proposed to improve the representation of the vectors based on the task at hand. As the humor related tasks involve common sense knowledge in addition to mere representation, we use the following semantic lexicon to improve the representation of word vectors.

2.2.1 Concept Net

ConceptNet is a semantic network containing lots of things computers should know about the world, especially when understanding text written by people. It is built from nodes representing words or short phrases of natural language, and labeled relationships between them. (We call the nodes "concepts" for tradition, but they'd be better known as "terms".) These are the kinds of relationships computers need to know to search for information better, answer questions, and understand people's goals.

3. PROPOSED METHOD

In the voiced model of speech, vocal tract transfer function (VTTF) values can be obtained only at the harmonics, then the problem of recovering the full VTTF at all frequencies can be seen as an interpolation problem. The number of harmonics are not constant and keep varying, so we decided to split the spectrum into fixed number of bands and treat maximum in each sub-bands as sample of VTTF. The recovery of full VTTF from these sub-band maximum values is explained below. The given speech signal is analyzed pitch adaptively using a Hanning window of 3 pitch periods. In unvoiced regions a constant window length of 15ms is used. The Voiced/Unvoiced decision is obtained using F0 contour (in our case STRAIGHT F0 was used but any F0 estimation[?] and voiced/unvoiced algorithm[?] will work). Then the MSASB procedure is applied to extract the spectral envelope from the short-time Fourier magnitude spectrum. The procedure is depicted in The first step is to compute the magnitude spectrum of the windowed signal by discrete Fourier transform. Then each spectral slice/frame is split into Nb sub-bands. These subbands are non-overlapping and of fixed bandwidth. The bandwidth of each sub-band can be found as fs/2*Nb. The maximum in each sub-band is then computed and stored, in addition the values of magnitude spectrum at 0,fs/2Hz are also stored. This is done for a constant frame shift of 5ms. The procedure is depicted in Fig. 1[?]

3.1 Results of Initial Experiment and Inference

From the table 1, it is apparent that mere word representations do not serve as useful tools for discrimination of humorous and non humorous data. In order to gain insight, we clustered the humor and non-humor words separately and observed that although these two classes have center of densities towards opposite ends, on an average, they form a non-separable cluster. This may be due to high volume of common data across the classes. Though we have huge amount of common words across these classes, we cannot remove them as they form an integral part to generate humor and unexpectation.

Also, single word vector models are severely limited since they do not capture compositionality. The dominant approach for building representations of multi-word units from single word vector representations has been to form a linear combination of the single word representations, such as a sum or weighted average. These approaches can work well when the meaning of a text is literally "the sum of its parts", but fails when words function as operators that modify the meaning of another word which usually happens in the case of humor. This inference leads to the clarity that we cannot use a linear classifier or word-level models for this task and is proved by the experiments we conducted using SVMs and Multi-layered Perceptrons. Hence we investigate the performance of deep neural networks like RNN(Recursive neural Networks) to achieve a non-linear separation.

Table 1: Classification Accuracy of one-liners vs Biography

Exp	SVM	MLP
Accuracy	27.15	29.16

3.2 Recursive Neural Network

In this section, we provide a brief intorduction to Recursive Neural Networks. Unlike standard neural networks, recursive neural networks (RNNs) are able to process structured inputs by repeatedly applying the same neural network at each node of a directed acyclic graph(DAG). In the past they have only been used in settings where another (often symbolic) component was first used to create directed acyclic graphs. These DAGs were subsequently given as input to the RNN. In such a setting, each non-leaf node of the DAG is associated with the same neural network. In other words, all network replications share the same weights. The inputs to all these replicated feed-forward networks are either given by using the child's labels to look up the associated representation or by their previously computed representation.

The models alternate between two stages:

- (1) Forward pass – recursively construct morpheme trees (cimRNN, csmRNN) and language model structures (csmRNN) to derive scores for training examples, and
- (2) Back-Propagation pass – compute the gradient of the corresponding objective function with respect to the model parameters.

3.3 Semantic Compositionality using Matrix Vector Recursive Neural Network (MV-RNN)

In the curent context , compositionality is the ability to learn compositional vector representations for various types of phrases and sentences of arbitrary length. The vector captures the meaning of that constituent. The matrix captures how the model modifies the meaning of the other word that it combines with. A representation for a longer phrase is computed bottom-up by recursively combining the words according to the syntactic structure of a parse tree. This model provides a new possibility for moving beyond a linear combination through use of a matrix W that multiplied the word vectors (a, b), and a nonlinearity function $g(.)$ (such

as a sigmoid or tanh). This function is recursively applied inside a binarized parse tree so that it can compute vectors for multi-word sequences.

The advantages of MV-RNN are:

- Assigns a vector and a matrix to every word
- Learns an input-specific, nonlinear, compositional function for computing vector and matrix representations for multi-word sequences of any syntactic type.

Assigning vector-matrix representations to all words instead of only to words of one part of speech category allows for greater flexibility which benefits performance. If a word lacks operator semantics, it's matrix can be an identity matrix. However, if a word acts mainly as an operator, such as "sadly", it's vector can become close to zero, while its matrix gains a clear operator meaning.

3.4 Improving Lexical Embeddings with Semantic Knowledge

The word embeddings obtained may not capture the desired semantics required and hence it makes sense to impart prior knowledge from semantic resources to learn improved lexical semantic embeddings. Suppose we have a resource that indicates relations between words. In the case of semantics, we could have a resource that encodes semantic similarity between words. Based on this resource, we learn embeddings that predict one word from another related word. We use the Relation Constraint model based joint training approach proposed in [**?**] with ConceptNet and WordNet as resources to improve the obtained embeddings.

4. DATASET

The dataset that we have used for our experiments were from the TIMIT database.It consists of 630 speakers.10 sentences of ecah speaker were collected and 80 percentage of the data was used for training and 20 percentage for testing.

5. EXPERIMENTS

We've analysed the performance of the proposed framework against the previous mentioned frameworks, using the same data as mentioned.

5.1 Experiment 1: systems built

5.1.1 Baseline

Using the Mfcc features extracted from the speech signals of 630 speakers, Gaussian Mixture Models have been developed.The results of using various centres are shown in the table.

5.1.2 MSASB on GMM

5.1.3 MSASB and Mfcc on GMM

5.1.4 MSASB on DNN

5.1.5 Mfcc and MSASB on DNN

- s1 sexual-content and semantic ambiguity
- s2 sexual-content, semantic ambiguity, and polarity

Table 2: Classification Accuracy of Funny vs Informative

Exp	Bayes	SVM	REPTree	MVRNN
s1	57.15	57.16	57.16	78.55
s2	57.35	57.38	57.36	82.46
s3	58.03	57.38	57.29	79.23
s4	58.26	57.94	58.31	78.10

Table 3: Classification Accuracy of Funny vs Insightful

Exp	Bayes	SVM	REPTree	MVRNN
s1	62.19	62.25	62.25	90.14
s2	62.66	62.43	62.74	88.78
s3	62.39	62.52	62.92	81.55
s4	63.08	62.97	63.52	87.24

- s3 sexual-content, semantic ambiguity, polarity and emotions
- s4 all features

All classifications experiments consider the classes funny versus informative, insightful, and negative respectively. The Tables 2-4 comprise the results.

5.2 Experiment 2

5.2.1 Top 5 MSASB and DNN

5.2.2 Agglomerative Information Bottleneck algorithm

The authors of [**?**] have made the data publicly available and hence we could run the evaluation on the same data. The authors have used Gradient Boosted Regression Trees for classification.

We can clearly see from the results in Tables 2-4 and 5 that Matrix-Vector Recursive Neural Networks using embeddings outperfrom baseline. The reason for this maybe due to the nature of sentences. The temporal information is highly vital to represent the semantics of a sentence and it is well captured and modeled by MV-RNN. This also can be attributed to the sematic relations captured inherently in the continuous space distribution of the Matrix Vector model of Recursive Neural Network. As humor is a highly cognitive aspect,it may not depend not just on a mere set of features/word representations, but rather on the relation between the features/words. Therefore, models like MV-RNN which capture the latent relationship between a pair of words intuitively perform better compared to a set of features. However, the features mentioned in [**?**] can be used as constraints while training the model itself to improve the performance.

6. CONCLUSION

This paper presents the prominance of usage of MSASBs in Speaker recognition tasks. We tabulated a series of experiments performed on MSASBs,MFCCs and I vectors. This work shall be further extended for bringing better results in speaker recognition.

Table 4: Classification Accuracy of Funny vs Negative

Exp	Bayes	SVM	REPTree	MVRNN
s1	60.37	60.36	60.37	92.54
s2	60.54	60.41	60.54	88.14
s3	60.13	60.37	60.54	91.44
s4	60.48	60.89	60.89	88.25

Table 5: Classification Accuracy of GBRT vs MVRNN

Exp	GBRT	MVRNN
Accuracy	0.817	0.824
F1	0.812	0.810