International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 4 (2012) pp. 619 - 627

Sentiment Analysis of Call Centre Audio Conversations using Text Classification

Souraya Ezzat[1], Neamat El Gayar[2], and Moustafa M. Ghanem[3]

[1]Center for Informatics Science, Nile University. Giza, Egypt
souraya.ezzat@nileu.edu.eg

[2]Center for Informatics Science, Nile University. Giza, Egypt
nelgayar@nileuniversity.edu.eg

[3]Department of Computing, Imperial College. London, England
mghanem@nileuniversity.edu.eg

Abstract: **The field of text mining has evolved over the past few years to help analyze the vast amount of textual resources available online. Text Mining, however, can be used also in various other applications. In this research, we are particularly interested in performing text mining techniques over transcribed audio recordings in order to detect the speakers' emotions. Our work is originally motivated by use cases arising from call centers, but can also have applications in other areas. We describe our overall methodology and present our experimental results for speech-to-text transcription, text classification and text clustering. We also focus on analyzing the effects of using different features selection methods.**

***Keywords*:** Sentiment Analysis, Audio and Text Mining, Feature Extraction and selection, Machine Learning, Call Classification and clustering.

I. Introduction

Whereas the use of automated sentiment analysis of textual documents has gained popularity over the past years, the application of audio sentiment analysis is still an open and challenging research area. Our long-term motivation is to investigate the development of novel methods that can integrate sentiment analysis of audio signals with the analysis of text that is automatically transcribed from recordings of human conversations.

Automated sentiment analysis itself is indeed useful for a variety of applications and is a vast topic of interest. It can empower interaction between humans and computers by enabling them to communicate in a more natural way [1]. For example, imagine computers that would be able to perceive and respond to human non-verbal communication such as emotions instead of responding conventionally to the events created by the use of a mouse or keyboard. In such a case, after detecting users' emotions, computers could personalize the settings according to the users' needs and preferences. In contrast to the traditional methods, computers could be sensitive to humans' emotions and take into account others means of humans' communicative power, for example especially those of handicapped individuals with disabilities.

Over the past few years, various research efforts have been devoted to transforming audio materials to text, for example songs, news, political debates to better integrate acoustically impaired individuals into society and involving them in daily activities such as listening to music, television programs, and movies. Other research work has investigated audio analysis to study customer-service/helpdesk phone conversations or even voice mails, others preferred to concentrate on text analysis as emotions are now frequently expressed on the internet through blogs, products/movies' reviews and news comments [2]–[3]–[4].

The approach we follow in this paper investigates the challenges and details of a simple conceptual approach for performing audio sentiment analysis over speech recordings, namely we first use automatic speech recognition tools to transcribe the recordings and then use text-based sentiment analysis approaches. In particular in this work, we contribute to the field by proposing a novel model that combines speech recognition technologies to analyze sentiment from calls using text classification methodologies to gain a better insight and a deeper understanding of the audios by analyzing their content. Our objectives are to classify calls according to sentiment (i.e positive or negative) in order to help call centre quality controllers monitor the performance of their agents, and get useful feedback about the satisfaction of their customers. Moreover, we extract a list of key words that differentiate positive versus negative calls in order to train future agents to detect instantaneously an unsatisfied customer and respond

appropriately to resolve his complaint immediately. Finally, we automatically cluster calls into similar groups, in an attempt to obtain other natural groupings not only positive and negative.

This model is a generic model that can serve many other purposes. For example, it can be applied in medical centers by helping psychiatrics diagnose their client's case through content analysis of their patients' speech recorded during treatment sessions or in analyzing politicians' oral debates. Generally, any audio material with emotionally deviating content or mood sways could be analyzed by first, converting it to text. The text can then be processed to extract features that discriminate between various emotional stages of the speaker and detect the overall sentiment in the conversation.

In this study we also exploit, several supervised and unsupervised machine learning techniques to analyze sentiments in calls. Different features generated from the converted text are investigated. Results are presented for a data set that has been replicated to simulate true conversations from a real call center. Although the experiments are still preliminary as they are based on a syntactic data set; we still yield interesting results and can draw useful conclusions.

The next section describes relevant background knowledge and reviews some related work previously conducted in the field. Section 3 carries on with an overview of the proposed model, followed by Section 4 which provides us with a description of the dataset and experiments. In Section 5, results are presented and discussed. Finally, the paper is concluded in Section 6 and future work is proposed.

II. Background and Related Work

A. Sentiment Analysis and Text Mining

Text mining refers to the process of deriving information from text through means such as statistical pattern learning by using parsing techniques to derive linguistics features. Typical text mining tasks include text categorization, text clustering, concept/entity extraction, document summarization and sentiment analysis [5]. Sentiment analysis typically aims to determine the attitude of a speaker with respect to some topic and deduce his emotional state accordingly. Sentiment analysis is generally more difficult than other text mining tasks [4].

A simple form of sentiment analysis is learning to classify whether documents express positive or negative sentiment. The task is generally trickier than that of traditional document classification. The author's writing skills and style can be subjective in a document. He might be criticizing ironically by using positive terms but intending the opposite. The majority of existing work on sentiment analysis has focused on supervised learning of a binary classifier using methods such as decision tree, naive Bayesian, maximum entropy, and support vector machine methods [2]–[3]–[4].

Other work on sentiment analysis use methods based on the sentence level analysis. For example, the work in [6],[7], calculates the "semantic orientation" of terms or phrases via their difference in mutual information with the words "excellent" and "poor" to avoid the necessity of a labeled training set. The work in [8] tries to label the sentences in each document as subjective and objective (discard the objective part) and then apply classical machine learning techniques for the subjective parts aiming to prevent the polarity classifier to consider irrelevant or misleading terms. However, this approach is not easy to test since it could be time consuming to collect a labeled data on the sentence level. Additionally, it requires heavy semantics pre-processing and pre-defined rules assigned upon contextual valence to the linguistics components.

B. Sentiment Analysis and Audio Mining

Recent researches on emotions recognition focuses mainly on extracting features from audio or visual information separately and more recently on a combined approach. All studies in the field point to the pitch (the fundamental frequency) as the main vocal cue for emotions recognition combined with vocal energy obtaining an accuracy statistics ranging from 65% to 88% [1]. Additional acoustic variables contributing to vocal emotion signaling are Mel-frequency cepstral coefficients (MFCC), Mel-based speech signal power coefficients, formants and temporal features such as speech rate and pausing [1]–[15]. Other approaches include derivative features such as LPC (linear predictive coding) parameters of signal. These features are mainly derived from short segments or utterances of speech and not from an entire continuous conversation. Particularly, some researchers attempt to combine visual information such as the region of eyes and eyebrows along with the previous acoustics features mentioned above to build a more robust emotional system [1]. However, in real life, it is difficult to obtain emotional video records as it requires additional tools and specifications. For instance, it would be merely impossible to capture on video of a furious customer complaining about an issue in a call center, unless he is a professional actor.

III. Proposed System: Analyzing Speech Content

In this paper we propose a model for speech analysis that utilises features extracted from the semantics of the signal to detect the emotions of the speakers. This model can basically be applied to analysing telephone calls and automatically rating them, for example in customer service call centres. We investigate the success of the approach in three tasks: 1) Classifying calls as showing positive/negative sentiment; 2) Extracting key words that differentiate positive versus negative calls; and 3) Automatically clustering calls into similar groups.

A. Emotions Recognition of Speech through Text

Figure 1 illustrates step by step how we aim at classifying the calls starting with the recorded audio files. Our approach is based on using speech recognition technology to automatically

generate text transcripts from the audios and then analyzing the transcribed texts using text mining technologies. As follows, we describe the steps for our proposed approach in more details.

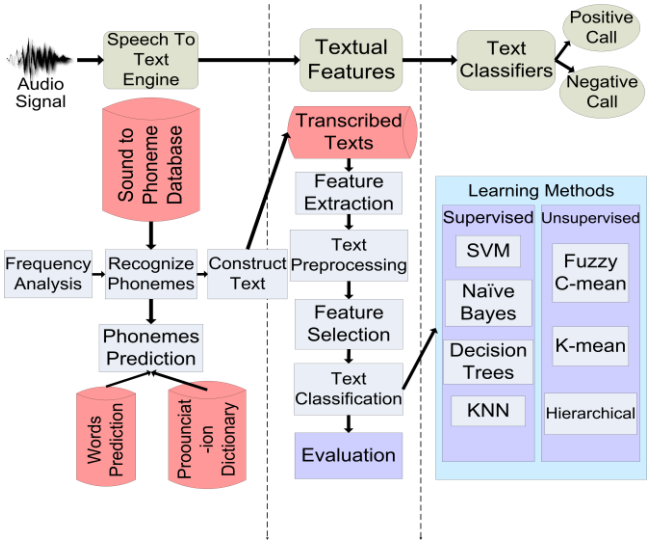

Figure 1. Text Classification of Speech Content Model

First the speech to text engine analyzes the frequency of the input sound wave. It then tries to match the sound to a phoneme and recognize several phonemes to predict the word in question and consequently, construct the entire conversation. The output generated is in form of a text file containing the dialogue between the agent and the customer.

Second, the transcribed text goes through a series of explorations. To start, we perform feature extraction by treating all documents' words as features and transforming the text into a "bag-of-words" representation, where each feature is represented by a single token. Each current word in the documents is then a candidate feature, but further preprocessing on these candidates is required to omit the most irrelevant ones and only keep the most important features, i.e. the ones to classify upon. Simple preprocessing techniques such as punctuation erasure, filtering out stop words/numbers, and stemming are used. Moreover, to avoid the curse of dimensionality, some feature selection metrics are computed, based upon which we reduce the feature set until no further removal increases error significantly. In particular, we investigate three different algorithms to extract relevant keywords: the keygraph based approach described in [9], the Chi^2 Keyword Extractor which extracts relevant keywords using co-occurrence statistics as described in [10] and by computing the Term Frequency TF and the Inverse Document Frequency IDF for each term and selecting the highest weight by defining a threshold.

Finally, the calls are classified using supervised and unsupervised learning techniques based on the feature vector selected. The rationale is that this approach would identify, and automatically learn, groups of words that are associated with service dissatisfaction and generate a concise human understandable list of words containing descriptive positive and negative terms.

B. Challenges

In contrast to traditional linguistics approaches of sentiment analysis, several challenges emerge when analyzing telephone calls. These calls are transcribed into texts but they do not constitute well-written rich vocabulary substances strongly indicating the sentiment of the author, but rather everyday speech used by normal people over the phone. Additionally, there might be some loss in translation or appearance of new words due to the poor telephone-quality. The presence of this additional noise makes the classifiers' task more challenging to detect the emotions of the speaker. Hence, we study the effect of different feature selection algorithms on noisy text due to the inaccuracy of speech recognition engines.

Moreover, the dataset collected for our experiments is not large enough to capture statistically robust information and automatically discovering patterns related to service satisfaction/ dissatisfaction in a real call centre. The reference scripts are hypothetically formulated and do not correspond to actual calls with manifested angry customers.

Furthermore, emotions can sometimes be hard to detect in a dialogue, since they are expressed through many other different forms, such as hand gesture, body language, voice tonality, laughers, physiological and brain activities, not necessarily with the use of a specific word or vocabulary. Additionally, it may vary from one individual to another. For example, some might be expressing their dissatisfaction with the use of abusive language, while others might prefer to be ironic by using positive terms to express the same dissatisfactory feeling while maintaining a calm tone. Therefore, selecting the best textual features for emotions recognition is a tricky task that should be handled carefully.

IV. Data and Experiments

A. Data Collection and Preprocessing

Our dataset consists of 36 audio files recorded in a controlled environment. We designed 12 different scenarios simulating real agent/customer conversations inspired from a call centre in Egypt. These scenarios are recorded and converted from speech to text to test the speech engine recognition performance. Consequently, the collection of transcribed text is mined. Twelve actors are involved in these recordings, 3 males and 9 females. The engine is tested on a general untrained profile. The scripts are pre-labeled depending on the scenarios, where 19 are marked positive denoting a successful call where the customer is satisfied from the service and 17 are labeled as negative, expressing customer's dissatisfaction.

B. Experimental Setup

Our approach uses language and discourse information, we explore the fact that some words are highly correlated with specific emotions. In this work we investigate four classifiers that are reported in the literature to work well for text

classification problems: the Support Vector Machines (SVMs), the Naïve Bayesian classifier, decision trees and the K-Nearest Neighbor classifiers [16, 17]. **SVMs** discriminate between two classes by fitting an optimal separating hyper-plane in the midway between the closest training samples of the opposite classes in a multi-dimensional feature space. We use a **SVM** model with a polynomial of 3rd degree with a bias and gamma of 1.0 as Kernel function parameters. The **Naïve Bayesian** classifier on the other hand uses Bayes rule and work under the assumption of conditional independence where each individual feature is assumed to be an indication of the assigned class, independent of each other. A naive Bayes classifier constructs a model by fitting a distribution of the number of occurrences of each feature for all the documents. In addition we also use a **Decision Tree (DT) Classifier.** Decision Trees are sometimes preferred over more accurate classifiers because of their descriptive power; i.e. the ability to interpret classification rules produced by the model. Finally the **Nearest Neighbor** algorithm classifies an object based on majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors. In this experiment, we choose k = 5, which is determined empirically using cross validation.

For the unsupervised methods, we select some of the most popular and efficient clustering techniques: the K-means, the fuzzy K-means and the Hierarchical Clustering [18]**.** The **K-means** performs crisp clustering assigning a data vector to exactly one cluster at a time. In contrast the **fuzzy C-means**, allows each data point to belong to several clusters with a degree of membership. On the other hand **Hierarchical Clustering** develops a sequence of partitions where at each step sub-clusters are combined according to some criterion. We use a hierarchical clustering algorithm with an Euclidean distance function and single linkage. We use KNIME open source data mining platform [13] for our experiments on classification and clustering. Results are presented and discussed in section 5.

C. Evaluation Metrics

To evaluate the performance of the automatic speech recognition engine, several metrics are computed to test how accurate the recognizer is able to retrieve the spoken words. By nature of the application, the speech recognition engine attempts to recognize each spoken utterance and predict it by mapping it to a word in its dictionary, so the order of the retrieved words is crucial in this specific application. Therefore, a dynamic programming alignment algorithm tries to find the optimal solution based on the lowest cost of operations. The algorithm seeks to find the longest common subsequence between the hypothesized text (HYP) and the reference text (REF) since the cost of correct match is zero. In this paper, the results are reported using *Sclite*, a tool for scoring and evaluating the output of speech recognition systems. *Sclite* is part of the *NIST SCTK* Scoring Toolkit[11]. The program compares the HYP output by the speech recognizer to the correct text by mapping all the correctly transcribed words to the REF. After comparing REF to HYP, statistics are gathered during the scoring process and the word error rate, also known as WER is computed using (1).

$$WER=(S+D+I) / N. \quad (1)$$

where 'S' is the number of substitutions, 'D' is the number of deletions, 'I' number of insertions and 'N' the total number of words in the reference text. Additionally, the word recognition rate, WRR, is calculated as in equation (2).

$$WRR=1-WER. \quad (2)$$

V. Results

A. Results of the Automatic Speech Recognition Engine

First, we transcribe the audio files into text to perform the text classification. We test *Dragon Naturally Speaking* engine [12] by computing the WER (word error rate) of each transcribed output. Results are presented in Table 1, 2 and 3. Table 1 compares the performance of different couples recording the same scripts. On the other hand, table 2 lists the performance of the same couple over different scenarios. Finally, table 3 summarizes the overall engine performance.

Table 1. Percentage of Word Recognition Rate (WRR) for DataSet1

Agent	Customer	script 1	script 2	script 3	script 4	Av/ Couple (%)
♂[1]$_1$	♂$_2$	28.7	40	47.2	36.8	38.17
♀$_1$	♀$_2$	52.1	61.4	56.9	57.2	**56.9**
♀$_3$	♀$_4$	40.3	41.4	49.2	45.5	44.1
♀$_5$	♀$_6$	50.6	42.9	53.4	48.8	48.92
♀$_6$	♀$_5$	44.6	44.3	45	47.8	45.42
♀$_2$	♂$_1$	44.9	52.5	53.9	45.9	49.3
♀$_3$	♀$_7$	35.5	48.9	51.1	44.5	45
Average/ Script		35.8	40.4	**42.9**	40.3	**39.88**

[1] ♂: male actors/ ♀: female actresses

Table 2. Percentage of Word Recognition Rate (WRR) for DataSet2

Agent/Customer	Script	WRR(%)
♀8 / ♀9	Script 5	47.9
	Script 6	42.2
	Script 7	31.9
	Script 8	35.6
	Script 9	34.2
	Script 10	**51.9**
	Script 11	38.3
	Script 12	18.7
	Av/Couple(%)	**37.21**

Table 3. Performance of Dragon Naturally Speaking speech engine

	Av. Performance (%)
Female Records	46.51
Male Records	38.17
Mix Records	47.15
Overall Performance	**43.9**

From the experiments we observe that the speech engine is gender dependent. Entirely female conversations are better recognized with 46.51% accuracy opposed to entirely male conversations with only 38.17% accuracy. Mixed dialogues achieve better performance with 47.15% accuracy. Additionally, the speech engine is speaker dependent. For example, when female5 plays the role of the agent versus female6 as the customer, they reach an average performance of 48.92% but when they switch roles their performance degrades and they only achieve 45.42%. Moreover, the engine is also context dependent since according to the script being acted different couples achieved better results for *scenario3* with 42.95% against 35.83% for *scenario1*. We can conclude that the speech engine is speaker dependent (gender, accent…), and also vocabulary dependent.

The overall average performance of the engine is approximately only 44% as shown in *table 3*. To increase this average performance we suggest transcribing the audios on a trained profile depending on the speaker's gender and train the language model to better recognize the context of the conversation.

B. Results of Features Extraction and Selection

In order to detect the terms that we believe are associated with service satisfaction or dissatisfaction, we use three different algorithms to extract the most common keywords present in our collection of transcribed text. Table 4 lists the top 12 words using the TF*IDF, the Chi^2 and the keygraph selection metric for the negatively labeled calls. It is obvious that the terms selected from negative calls indicate complaints about delays and payments from previous calls related to domain specific issues. The client often refers to a previous call, "*yesterday*'s *call*", or asking about a reimbursement or a warranty. Interestingly, the word "*sorry*" appears distinctively in calls where the client is dissatisfied from the service, and hence the agent is obliged to apologize for a mistake they did in order to calm him down. In contrast, terms selected from positive calls, as listed in table 5, show courtesy and consideration. For example, the terms "*thank you*, *please*, *okay* and *yeah"* are used extensively to reflect the mutual agreement and accord between the agent and the customer during their conversation. It can be noted from Tables 4 and 5 that the keywords appearing depend on the selection algorithms. However, some keywords remain common amongst all feature selection methods and hence relate to the characteristics of the call.

Table 4. Top Terms Derived from Negative Calls using Different Selection Metrics

Terms Selected from Negative Calls		
TF*IDF	Chi^2	Keygraph
told	told	told
call	call	call
time	time	time
pay	pay	Pay
warranti	warranti	warranti
sorri	sorri	Month
hour	hour	Reason
reimburse	reimburse	do
speak	game	game
yesterdai	replace	*yesterdai*
charge	expect	refer

Table 5. Top Terms Derived from Positive Calls using Different Selection Metrics

Terms Selected from Positive Calls		
TF*IDF	Chi^2	Keygraph
okai	okai	okai
check	check	check
thank	thank	thank
please	please	please
account	account	account
yea	yea	refer
name	name	power
email	dai	dai
website	send	website
proceed	mean	mean
double check	suppli	moment

C. Results of Text Classification

In this section we present the results of text classification on the call center data described in section 4.1. Figures 2, 3 and 4 compare the classification accuracy of the SVM, the Naïve Bayesian classifier, the Decision Tree classifier and the K-Nearest Neighbors classifiers using different feature selection algorithms. Since we use cross-validation to evaluate the performance of these classifiers, in each iteration the training and test sets are different. Hence the features extracted from the training documents also vary depending on the partitioning results. Our target class of interest for this specific application is the "negative" class where unsuccessful calls occur.

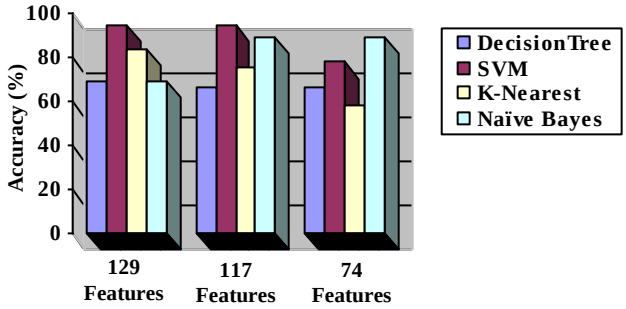

Figure 2. Classification based on Keygraph Keyword Extractor.

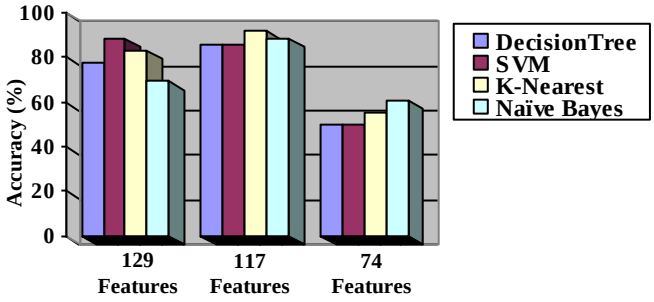

Figure 3. Classification based on TF*IDF Keyword Extractor.

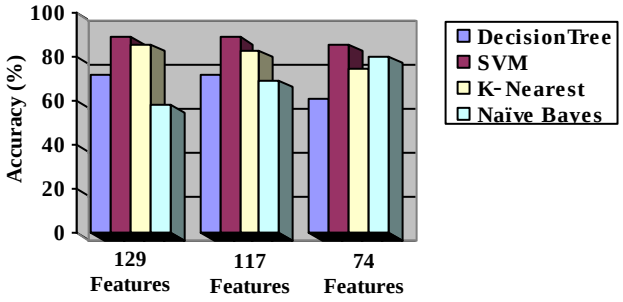

Figure 4. Classification based on Chi^2 Keyword Extractor.

By inspecting figures 2-4, it can be observed that as we reduce the features set, the Naïve Bayes seems to perform better, except for the TF*IDF selection technique. On the other hand, SVM and K-Nearest achieve higher accuracies with larger set of features. Finally, the decision tree performance seems not to be affected by the feature selection methodology, besides its performance seems to be the worst amongst all.

Table 6 summarizes the best performance of the 4 different classifiers used and lists the accuracy, true positives, false positives, true negatives, false negatives, along with precision and F-score of the models. Refering to Table 6, it can be stated that the SVM with Keygraph extractor outperform the rest of the classifiers. The decision tree has the worst performance amongst all with only 86.1% accuracy. The support vector machine model only misclassifies 2 negative instances as positive. The Keygraph algorithm for keyword extraction achieves an overall good performance and the highest performance for the SVM and Naïve Bayesian classifiers. However, using TF*****IDF score as term weighting, still achieve good results particularly with the Naïve Baysian classifier, the Decision Tree and the K-Nearest neighbor.

Table 6. Classifiers' Best Performance

	Decision Tree TF*IDF	**Naïve Bayesian KeyGraph/ TF*IDF**	**SVM Key Graph**	**K-Nearest TF*IDF**
Classification Accuracy (%)	86.1	88.9	**94.4**	91.7
Sensitivity	0.889	0.826	1	1
Specificity	0.833	1	0.90	0.864
True Positives	15	19	15	14
False Positives	2	0	2	3
True Negatives	16	13	19	19
False Negatives	3	4	0	0
Precision	0.882	1	0.88	0.824
F-measure	0.852	0.905	0.93	0.927

D. Results of Clustering

In this section we present the clustering results of our collected data set which consists of 36 transcribed text documents. Here the purpose of this experiment is to group similar conversations (i.e. documents) w.r.t some characteristics hoping that we can uncover positive from negative calls without the need of labeling. In fact, that would considerably facilitate the job for quality assurance personnel. Huge amount of unlabelled data coming from thousand of recorded calls every day are recorded daily, and only few of them are being monitored manually. The proposed approach will tempt to automate finding natural grouping and store them together. These groups will later be analyzed. Again we use KNIME's platform to compare the performance of the hierarchical clustering, the K-mean clustering and the Fuzzy C-mean clustering algorithms. Each time, we test the models

with different set of features extracted from the TF*IDF, the Chi^2, and the Keygraph extraction algorithms [13]. In our initial experiment we choose the number of clusters to be 2. However, we study the effect of reducing the features set, by only selecting the top ranking keywords in the collection of documents as discriminative features. Results are depicted in figures 5, 6 and 7. Clustering results only group similar patterns (i.e. here text transcribed from calls). To be able to have an insight of how good clustering can separate positive from negative calls we use the labeled data set to first label each cluster according to the labels of the majority of the patterns that belong to this cluster. We can then calculate the performance of the clustering algorithm in terms of accuracy. The accuracy is depicted in figures 5-7 so one can also compare it to the classification results.

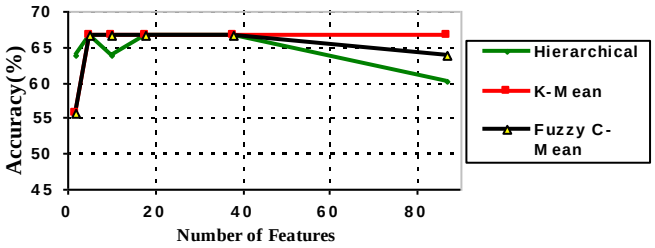

Figure 5. Clustering based on TF*IDF Keyword Extractor.

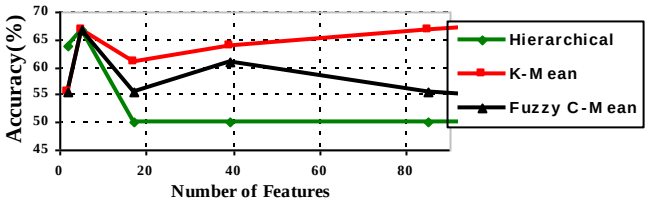

Figure 6. Clustering based on Keygraph Keyword Extractor.

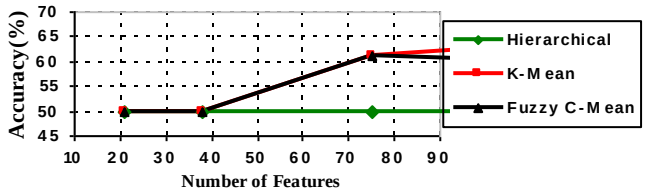

Figure 7. Clustering based on Chi^2 Keyword Extractor.

For the three different extraction approaches, the K-Means algorithm seems to better perform than the fuzzy C-means and the hierarchical clustering. Each method reaches its peak performance at 66.7% of correct classification but with different number of features depending on the specified threshold of selection. The hierarchical clustering has a very poor performance with only 50% of correct classification, since it predicts all the text in one cluster and hence, it has a 50% chance of being correct.

As an observation, from figure 6, Keygraph seems to work better with low number of features, as opposed to Chi^2 which performs better with larger number of features. On the other hand, TF*IDF keyword extractor method works with a more or less reliable- performance regardless of the number of features being added. Therefore one can state that the TF*IDF is a more robust metric selection algorithm. This also agrees with the classification results previously presented and summarized in Table 6.

Moreover, we study the effect of changing the number of clusters while maintaining the same number of features for the three different clustering methods. We later attempt to analyze the clusters by looking deeper into the calls that were regrouped automatically together. Results are illustrated in figures 8, 9 and 10.

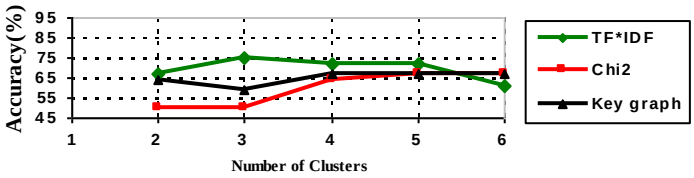

Figure 8. K-Means Clustering.

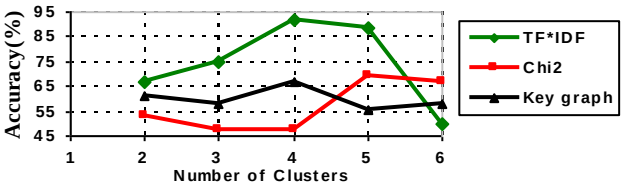

Figure 9. Hierarchical Clustering.

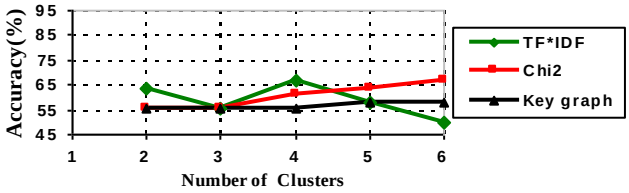

Figure 10. Fuzzy C-Means Clustering.

One can observe that when we attempt to group similar patterns into four different clusters we achieve better results than with only two clusters. The fuzzy C-means based on features extracted using TF*IDF method hits 91.3% of accuracy when we match cluster 0 and 1 against positive label and cluster 2 and 3 against negative label. These four clusters actually represent the main four scripts we have mimicked and recorded using different speakers. Indeed, two of these scenarios were positive and the other two were negative. However, the performance degrades when we further increase the number of clusters.

VI. Conclusion and Future Work

This work presents a general model that accepts any type of audio material and studies its content through machine learning techniques by automatically converting the audios into text and mining the text content. In this research, we propose several text mining techniques on recorded telephone calls mimicking real agent/customer conversations after translating them into text in order to detect the speakers' emotions, and hence predict whether the customer is satisfied or dissatisfied of the service provided. The proposed approach is based on using speech recognition technology to automatically generate those text transcripts and then subsequently analyzing them using various text mining technologies.

Although, the transcribed text being mined represents 44% of what is originally being said in the conversation, the different text mining techniques used achieve promising results. Therefore, highly accurate transcription may not be essential to provide us with good classification. Supervised learning techniques works generally well for text classification.

As stated before the study presented in this paper introduces some preliminary results on an artificially generated data set. We are currently collecting a larger dataset and will be extending our experiments on a wider scale. Additionally, we intend to train the speech recognition engine in order to enhance the recognition and improve the quality of text generated. Moreover, we intend to segment the original audios signal to obtain each speaker on separate segments, and hence study the change of emotions in the call. This new information would be added as an additional feature to help the classifiers learn better.

Finally, we are in the process of implementing a new model combining the previously discussed approach of text classification with another approach detecting the speakers' emotions from their acoustics features. We expect this model to compensate for the loss in translation and hence, improve the overall system performance by fusing both classifiers' decisions.

References

[1] Zhigang, Deng. "Analysis of Emotion Recognition using Facial Expressions, Speech and Multimodal Information". In *Proceedings of ACM 6th International Conference on Multimodal Interfaces (ICMI 2004)*, pp. 205-211, 2004.

[2] Bo Pang and Lillian Lee. "A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts". In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*. Introduced polarity dataset v2.0, subjectivity dataset v1.0. pp. 271-278, 2004.

[3] Bo Pang and Lillian Lee. "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales". In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics, (ACL* 2005). Introduced scale dataset v1.0 and a positive/negative sentence collection, 2005.

[4] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs up? Sentiment Classification using Machine Learning Techniques". In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP2002)*. Introduced polarity dataset v0.92002.

[5] Sajib Dasgupta and Vincent Ng. "Topic-wise, Sentiment-wise, or Otherwise? Identifying the Hidden Dimension for Unsupervised Text Classification". In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP2009)*. pp. 580-589, 2009.

[6] Stephen D. Durbin, J. Neal Richter, and Doug Warner. "A System for Affective Rating of Texts". In *Proceedings of the KDD Workshop on Operational Text Classification Systems*. October, 2003.

[7] Peter D. Turney. "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews". In *Proceedings 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, pp. 417-424, 2002.

[8] Mostafa Al Masum Shaikh, Helmut Prendinger and Mitsuru Ishizuka. "Assessing Sentiment of Text by Semantic Dependency and Contextual Valence Analysis". In *Proceedings of the Second International Conference on Affective Computing and Intelligent Interaction ACII* 2007 (LNCS 4738). pp. 191-202, 2007.

[9] Masao, Nakada and Yuko, Osana. "Document clustering based on similarity of subjects using integrated subject graph. International Association of Science and Technology for Development". In *Proceedings of the 24th IASTED international conference on Artificial intelligence and applications*, pp. 410- 415, 2006.

[10] Yutaka Matsuo and Mitsuru Ishizuka. "Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information". In *Proceedings of the 16th International Florida on Artificial Intelligence Research Society Conference (FLAIRS2003)*. pp.392-396, 2003.

[11] "The History of Automatic Speech Recognition Evaluations at NIST". National Institute of Standards and Technology. May, 2009. Retrieved May, 2010.

[12] Dragon Naturally Speaking Preferred Edition 10.0. Nuance. http://www.nuance.com/naturallyspeaking/default.asp

[13] Michael R. Berthold, Nicolas Cebron, Fabian Dill, Thomas R. Gabriel, Tobias Kotter, Thorsten Meinl, Peter Ohl, Christoph Sieb, Kilian Thiel, and Bernd Wiswedel. "KNIME: The Konstanz Information Miner. Studies in Classification, Data Analysis, and Knowledge Organization", Springer, 2007.

[14] Fabrizio, Sebastian. "Machine Learning in Automated Text Categorization". ACM Computing Surveys, Vol. 34, No. 1, pp. 1-47. March, 2002.

[15] Esraa Ali Hassan, Neamat El Gayar and Moustafa M. Ghanem , "Emotions Analysis of Speech for Call Classification", In *Proceedings of the 10th IEEE International Conference on Intelligent Systems Design and Applications (ISDA 2010)*, November 29-December 1, Cairo, Egypt, pp. 242-247, 2010.

[16] Alpaydin, E. Introduction to Machine Learning. The MIT Press, ISBN 0-262-01211-1. 2004.

[17] Duda, R. O., Hart, P. E. & Stork, D. G, Second edition. "Pattern Classification", John Wiley & Sons, 2000

[18] Witten, I.H., Frank, E. "Data Mining: Practical Machine Learning Tools and Techniques", Second Edition. Morgan Kaufmann, San Francisco, 2005.

Author Biographies

Souraya Ezzat was born in Cairo, Egypt. She had studied at Lycée Français du Caire. Graduated from the American University in Cairo with high honors, with a degree in Computer Science, in 2009. In 2010, she joined Nile University as a Research Assistant in the team of Data Mining and Machine Learning. While continuing her master degrees in Information and Communication Technology, expected to finish post-graduate studies in 2012.

Neamat El Gayar is currently a visiting scientist at the Centre of Pattern Recognition and Machine Intelligence (CENPARMI) in Concordia University, Canada. She is also an associate professor at Faculty of Computers and Information, Cairo University, Egypt. She obtained her Ph.D. and her M.Sc. in Computer Science from the Faculty of Engineering, University of Alexandria, Egypt. She joined the Nile University, Egypt for two years were she was the head of the data and data mining research group at the centre of Informatics Sciences. Dr. Neamat's research interests lie mainly in the fields of pattern recognition and machine learning, data mining and intelligent data-analysis techniques. She currently has over 50 refereed publications in these areas. Dr. El Gayar was appointed as vice chair of the IAPR Technical Committee 3 on Neural Networks & Computational Intelligence.

Moustafa Ghanem is Vice President for Research at Nile University and a Research Fellow at the Department of Computing, Imperial College London. He was previously the Research Director of the Imperial College spinout company InforSense Ltd, and the Founder and Co-Director of the Center for Informatics Science at Nile University. He holds a PhD and an MSc in high performance computing from Imperial College London and a BSc in electronics and telecommunications engineering from Cairo University. His academic research interests include large-scale informatics infrastructures; distributed data mining and text mining; grid and cloud computing middleware design and programming models and their applications in bioinformatics and other scientific applications. He has more than 70 academic papers published in these areas.