# IIIT-H at IJCNLP-2017 Task 4: Customer Feedback Analysis using Machine Learning and Neural Network Approaches

**Prathyusha Danda**[1]
IIIT-Hyderabad
Hyderabad, India
500032

**Pruthwik Mishra**[1]
IIIT-Hyderabad
Hyderabad, India
500032

**Silpa Kanneganti**[2]
i.am+ LLC.
Bangalore, India
560071

**Soujanya Lanka**[3]
i.am+ LLC.
37 Mapex building
Singapore - 577177

## Abstract

The IJCNLP 2017 shared task on Customer Feedback Analysis focuses on classifying customer feedback into one of a predefined set of categories or classes. In this paper, we describe our approach to this problem and the results on four languages, i.e. English, French, Japanese and Spanish. Our system implemented a bidirectional LSTM(Graves and Schmidhuber, 2005) using pre-trained glove(Pennington et al., 2014) and fastText(Joulin et al., 2016) embeddings, and SVM (Cortes and Vapnik, 1995) with TF-IDF vectors for classifying the feedback data which is described in the later sections. We also tried different machine learning techniques and compared the results in this paper. Out of the 12 participating teams, our systems obtained 0.65, 0.86, 0.70 and 0.56 exact accuracy score in English, Spanish, French and Japanese respectively. We observed that our systems perform better than the baseline systems in three languages while we match the baseline accuracy for Japanese on our submitted systems. We noticed significant improvements in Japanese in later experiments, matching the highest performing system that was submitted in the shared task, which we will discuss in this paper.

## 1 Introduction

Customer feedback analysis is a dominant problem, to the extent that there are companies whose principal purpose is to categorize feedback data. Classification of customer feedback would help companies gain a better perspective on the views of the customer. Comprehending customer feedback not only helps to understand the customer pulse better, but also to reply with an appropriate response. Hence, many companies understandably want an automated customer feedback analysis system. A major hurdle while doing this is dealing with the multilingual environment that is existent in most of the countries.

Considering the above points, the aim of the IJCNLP shared task on Customer Feedback Analysis is to classify real world customer feedback reviews into pre-defined set of classes. The goal is to achieve this by using data driven techniques in machine learning, which will help automate the classification process. The customer feedback are extracted, from Microsoft Office customers, in four languages, i.e. English, French, Spanish and Japanese. Since, there is no universal categorization for customer feedback, a set of six classes which would be applicable to all the entire set irrespective of the language they belong to, are created. These six classes are *comment*, *request*, *bug*, *complaint*, *meaningless* and *undetermined*. Each feedback was tagged with one or more classes. The task was to use this annotated data and build a model using supervised techniques. The model should be able to categorize a given review in one of the four aforementioned languages, into one or more of the classes.

We used bi-directional LSTMs (Graves and Schmidhuber, 2005) for the classification task at hand. We also used simple Naive Bayes classifier and SVM models as separate alternate approaches to achieve the intended goal. We found that the accuracy of the SVM model was almost on par with the bi-directional LSTM for English. We used glove pre-trained embeddings[1] (Pennington et al., 2014) for English while for the rest

[1] we used Common Crawl corpus with 840B tokens, 2.2M vocab, case-sensitive, 300-dimensional vectors available on https://nlp.stanford.edu/projects/glove/



*Proceedings of the 8th International Joint Conference on Natural Language Processing, Shared Tasks*, pages 155–160,
Taipei, Taiwan, November 27 – December 1, 2017. 

of the languages we used fastText[2] (Joulin et al., 2016) embeddings. The SVM model with TF-IDF as features performed better for French data compared to the bi-directional LSTM with fastText word embeddings. For Japanese and Spanish language data, bi-directional LSTM with fastText models have performed better compared to the SVM with TF-df models respectively. Both these models made use of fastText word embeddings of those particular languages. The Naive Bayes with TF-IDF (McCallum et al., 1998) models on all four languages gave lesser accuracies compared to the SVM (Cortes and Vapnik, 1995) and bi-LSTM models.

This paper is organized as the following - section 2 explains the related work and section 3 details about the corpus. Different approaches employed are explained in the subsequent sections. Results constitutes sections 5 and Error Analysis & Observation are presented in section 6. We conclude our paper with the Conclusion & Future Work section.

## 2 Related Work

(Bentley and Batra, 2016) dealt with Microsoft Office users feedback, on which they applied various machine learning techniques. They implemented classification techniques on labeled data and applied clustering approaches for unlabeled data. They had reported 0.5667 recall and 0.7656 precision on English data using logistic regression in a one-versus-rest setting. The text was lemmatized, stop words were removed and POS tags and named entities were tagged in the pre-processing stage. They used n-grams up to 3 in a bag-of-words approach along with non-text features such as star rating, sentiment and the categorization that an agent gave to the feedback, from the predefined Agent Taxonomy. Another key point of this particular model is that the users have scope to label the already predicted data, thus making rebuilding and re-predicting feasible with newer and larger data, which can increase the precision of the model.

The 2017 GermEval task (Wojatzki et al., 2017) has been designed to extract opinions from the customer feedback on the services offered by a German public train operator Deutsche Bahn. The shared task included 4 subtasks - relevance prediction, document polarity identification, aspect and category identification, target opinion extraction. SVM (Cortes and Vapnik, 1995) and CRF (Lafferty et al., 2001) have been used to create baseline models[3].

Sentiment analysis on customer feedback data by (Gamon, 2004) using linear SVM for classification had yielded satisfying results. They used feedback items from a Global Support Services survey and a Knowledge Base survey. Satisfaction scores were on a scale from 1(not satisfied) to 4 (very satisfied). Each feedback was given one of these 4 scores. They used surface level features like n-grams as well as linguistic features like POS tagging. All the features were represented in a binary form (present or not present) except for the length of sentence. Feature reduction was done based on log likelihood ratio. F1 measure of 74.62 was reported for 1 versus 4 classification and 58.14 for 1 and 2 grouped together versus 3 and 4 grouped together, both using top 2k features.

## 3 Corpus Details

The statistics of the corpus used for this task is detailed in Table 1. A few examples extracted from the training set are given below:-

- The sentence "Some are fairly easy, but I definitely get stuck." is tagged as a "comment".
- The sentence "The only thing that wasn't that perfect was the internet connection." is labeled as "comment, complaint".
- "All offered drinks and food at the restaurants and bars are too expensive." is a sentence with tag "complaint".

## 4 Approach

Many machine learning algorithms rely heavily on features designed by domain experts which makes the labeling task cost inefficient. So we have not used any language specific features like part-of-speech tag, morph features, dependency labels etc. for the task. We describe our approaches in the following subsections.

### 4.1 Machine Learning Approaches

We used Support Vector Machines (SVM), logistic regression(Log-Reg), k-Nearest Neighbor(k=3, 5 for our experiments) and Gaussian Naive

[2]https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

[3]https://github.com/uhh-lt/GermEval2017-Baseline

| Lang | Type | #Tokens-Case-Sensitive | #Tokens-Lowercase |
|---|---|---|---|
| English | Train | 5449 | 4674 |
| | Dev | 1848 | 1663 |
| | Test | 1788 | 1589 |
| Spanish | Train | 3524 | 3119 |
| | Dev | 1043 | 933 |
| | Test | 1109 | 1015 |
| French | Train | 3930 | 3515 |
| | Dev | 1454 | 1332 |
| | Test | 1407 | 1306 |
| Japanese | Train | 1651 | 1648 |
| | Dev | 282 | 281 |
| | Test | 320 | 320 |

Table 1: Corpus Statistics for the Shared Task

| Lang | Tag | Model | MicroAvg |
|---|---|---|---|
| English | bug | bi-LSTM | **0.19** |
| | | SVM | -1.0 |
| | comment | bi-LSTM | **0.81** |
| | | SVM | 0.80 |
| | complaint | bi-LSTM | **0.63** |
| | | SVM | 0.62 |
| | meaningless | bi-LSTM | **0.40** |
| | | SVM | 0.25 |
| | request | bi-LSTM | 0.13 |
| | | SVM | **0.29** |
| | undetermined | bi-LSTM | **-1.0** |
| | | SVM | **-1.0** |
| Spanish | bug | bi-LSTM | **-1.0** |
| | | SVM | **-1.0** |
| | comment | bi-LSTM | **0.92** |
| | | SVM | **0.92** |
| | complaint | bi-LSTM | **0.75** |
| | | SVM | 0.68 |
| | meaningless | bi-LSTM | **-1.0** |
| | | SVM | **-1.0** |
| | request | bi-LSTM | 0.50 |
| | | SVM | **0.53** |
| French | bug | bi-LSTM | 0.17 |
| | | SVM | **0.22** |
| | comment | bi-LSTM | 0.80 |
| | | SVM | **0.84** |
| | complaint | bi-LSTM | **0.61** |
| | | SVM | **0.61** |
| | meaningless | bi-LSTM | **0.44** |
| | | SVM | 0.36 |
| | request | bi-LSTM | **0.15** |
| | | SVM | -1.0 |
| | undetermined | bi-LSTM | **-1.0** |
| | | SVM | **-1.0** |

Table 2: Results on Test Data for English, Spanish and French using SVM with fastText features

| Tag | MicroAvg |
|---|---|
| bug | 0.4 |
| comment | 0.86 |
| complaint | 0.74 |
| request | 0.44 |
| undetermined | -1.0 |

Table 3: Updated Test Results for Japanese. SVM model here used unigram-bigram tf-idf vectors as features

Bayes(NB) using sklearn library (Pedregosa et al., 2011). In the ML approaches, we used TF-IDF (Sparck Jones, 1972) vectors for the words(uni)[4] present in the training corpus. We also experimented with TF-IDF vectors for bigrams(bi) and trigrams(tri). Sklearn uses count vectorizers to convert text input into a collection of tokens. It gives the flexibility of including higher n-grams in the vocabulary. This can prove to be helpful in the classification task. We used sklearn linear SVM library with the settings mentioned in Table 9. We employed the one-versus-one strategy for the classification task. We implemented Naive Bayes where all the features are assumed to follow Gaussian distribution. We also created a logistic regression model with maximum iteration of 100 and tolerance level of 0.0001.

### 4.2 Neural Networks

We implemented mainly two neural network models - bi-directional LSTM(bi-LSTM) (Graves and Schmidhuber, 2005), multi-layer perceptron (MLP) (Sparck Jones, 1972) using (Chollet et al., 2015). The accuracies of these models are reported in the subsequent sections. These neural network models used word embeddings as features. Glove embeddings were used for English and fastText embedding for other three languages. For encoding a sequence, bidirectional LSTM uses contextual information in both the directions - past and future word vectors. This enables them to have a better semantic representation of any sequential data. The maximum length of the sample was set to 100. We used word embeddings of size 300 for all the languages. For MLP, we used a single hidden layer of 300 nodes. The sentences which have more than 100 tokens would be truncated and only the first 100 tokens take part in the learning. Adam optimizer (Kingma and Ba, 2014) was used for learning with default learning rate of 0.001 and categorical cross-entropy loss function.

## 5 Results

The experimental results on the development and test data for different languages are shown in Tables 2-9. The highest performing system measures are marked in bold. From the tables 4, 6 and 7, it can be seen that SVM with a linear kernel and TF-IDF features outperforms all other machine learn-

[4] All the keywords written in parenthesis are later used in the tables.

| Lang | Model | Features | Exact-Accuracy | Partial-Accuracy | Macro-Average | Micro-Average |
|---|---|---|---|---|---|---|
| English | SVM | uni | 0.67 | 0.67 | **0.35** | 0.68 |
| | | uni-bi | 0.67 | 0.67 | 0.34 | 0.68 |
| | | uni-bi-tri | **0.68** | **0.68** | **0.35** | **0.69** |
| | | glove-vectors | 0.57 | 0.57 | **0.35** | 0.59 |
| | NB | uni | 0.67 | 0.67 | **0.35** | 0.68 |
| | 3-NN | uni | 0.55 | 0.55 | 0.25 | 0.55 |
| | 5-NN | uni | 0.55 | 0.55 | 0.23 | 0.56 |
| | Log-Reg | uni | 0.66 | 0.66 | 0.24 | 0.67 |
| Spanish | SVM | uni | **0.90** | **0.90** | 0.46 | **0.89** |
| | | uni-bi | 0.82 | 0.82 | 0.27 | 0.82 |
| | | uni-bi-tri | 0.82 | 0.82 | 0.27 | 0.82 |
| | NB | uni | 0.77 | 0.77 | 0.32 | 0.77 |
| | 3-NN | uni | 0.84 | 0.84 | **0.57** | 0.84 |
| | 5-NN | uni | 0.85 | 0.85 | 0.55 | 0.85 |
| | Log-Reg | uni | 0.88 | 0.88 | 0.32 | 0.88 |
| French | SVM | uni | 0.74 | 0.74 | **0.36** | 0.76 |
| | | uni-bi | 0.60 | 0.60 | 0.16 | 0.62 |
| | | uni-bi-tri | 0.60 | 0.60 | 0.16 | 0.62 |
| | NB | uni | 0.52 | 0.52 | 0.30 | 0.63 |
| | 3-NN | uni | 0.64 | 0.64 | 0.30 | 0.66 |
| | 5-NN | uni | 0.61 | 0.61 | 0.26 | 0.64 |
| | Log-Reg | uni | **0.88** | **0.88** | 0.32 | **0.88** |
| Japanese | SVM | uni | 0.70 | 0.70 | 0.59 | 0.72 |
| | | uni-bi | 0.73 | 0.73 | 0.62 | 0.75 |
| | | uni-bi-tri | **0.74** | **0.74** | **0.65** | **0.77** |
| | NB | uni | 0.38 | 0.39 | 0.31 | 0.44 |
| | Log-Reg | uni | 0.68 | 0.68 | 0.45 | 0.70 |

Table 4: Results on Development Data for English, Spanish, French and Japanese using ML Approaches. Updated models used for Japanese

| Language | Model | Features | Exact Accuracy | Micro-Average | Partial Accuracy | Macro-Average |
|---|---|---|---|---|---|---|
| English | MLP | glove | 0.63 | 0.63 | 0.36 | 0.65 |
| | SVM | fastText | 0.57 | 0.57 | 0.35 | 0.59 |
| | bi-LSTM | glove | **0.65** | **0.66** | **0.45** | **0.68** |
| Spanish | MLP | fastText | 0.81 | 0.81 | 0.32 | 0.81 |
| | SVM | fastText | 0.76 | 0.76 | 0.36 | 0.76 |
| | bi-LSTM | fastText | **0.86** | **0.86** | **0.42** | **0.86** |
| French | MLP | fastText | 0.66 | 0.66 | 0.33 | 0.69 |
| | SVM | fastText | 0.60 | 0.60 | 0.27 | 0.64 |
| | bi-LSTM | fastText | **0.71** | **0.71** | **0.38** | **0.74** |
| Japanese | MLP | fastText | 0.57 | 0.57 | 0.30 | 0.57 |
| | SVM | uni | **0.60** | **0.60** | 0.47 | **0.60** |

Table 5: Results on Development Data Using Neural Networks. SVM model here uses fastText vectors as features for English, Spanish and French, where as character unigrams for Japanese. Updated models used for Japanese

| Language | Model | Exact Accuracy | Partial Accuracy | Micro-Average | Macro-Average |
|---|---|---|---|---|---|
| English | bi-LSTM | **0.65** | 0.65 | **0.68** | **0.36** |
| | SVM | **0.65** | **0.66** | **0.68** | 0.34 |
| Spanish | bi-LSTM | **0.86** | **0.86** | **0.86** | 0.44 |
| | SVM | 0.85 | 0.85 | 0.85 | **0.45** |
| French | bi-LSTM | 0.65 | 0.65 | 0.69 | **0.38** |
| | SVM | **0.70** | **0.70** | **0.72** | **0.38** |
| Japanese | bi-LSTM | **0.56** | **0.57** | **0.56** | **0.28** |
| | SVM | **0.56** | 0.56 | **0.56** | 0.26 |

Table 6: Test Results; using older Japanese models. SVM model here uses TF-IDF vectors as features

| Lang | Model | Features | Exact-Accuracy | Partial-Accuracy | Macro-Average | Micro-Average |
|---|---|---|---|---|---|---|
| Japanese | SVM | uni | 0.74 | 0.74 | **0.52** | 0.75 |
| | | uni-bi | **0.76** | **0.76** | **0.52** | **0.77** |
| | | uni-bi-tri | 0.75 | 0.75 | 0.51 | 0.76 |
| | Log-Reg | uni | 0.67 | 0.70 | 0.45 | 0.70 |

Table 7: Updated Test Results for Japanese

| Parameter | Value |
|---|---|
| Loss | Squared Hinge Loss |
| Penalty | L2 |
| Iterations | 1000 |
| Tolerance | 0.0001 |

Table 8: SVM Parameters

| Class Pairs | #Common Words |
|---|---|
| Complaint Comment | 1155 |
| Meaningless Comment | 657 |
| Complaint Meaningless | 584 |
| Comment,Complaint Comment | 514 |

Table 9: Class Distribution of Overlapping English Training Data

ing techniques. Naive Bayes' classifier relying on the maximum likelihood estimates performed poor across languages. K-nearest neighbor algorithm which depends on the vector representation of feedback and the euclidean distance from other vectors also did not perform reliably which is evident from the tables. From Table-5, we observed that bidirectional-LSTMs outperformed MLP and SVM when word vectors were considered as features. Bi-LSTMs represent a sequence better than the other two which in-turn increases the classification accuracy.

## 6 Error Analysis & Observation

A major observation in the data is that many words overlapped between different classes. The maximum overlap is observed between comment and complaints and this contributes to many false positives. The meaningless tag adds to the confusion, as these sentences have a huge overlap with comment and complaint classes. As the "undetermined" tag was not present in the training data, the system was unable to predict it. The labels which are combinations of two atomic labels are also contentious ones. For example: the label "comment, complaint" gets confused with "comment" as well as "complaint". The partial accuracy metric captures this whether one label is matched when the true label consists of two labels. The top four overlapping classes for English are shown in the Table 9. The statistics were got on English data as it had the maximum training samples. Examples of test errors-

- For the sentences "Lunch, they forgot one meal.", the system wrongly predicts the tag as "comment" while the correct label corresponds to "complaint".
- The sentence "This editor is good, but could still use some Hot needed improvements!" is tagged as "comment, complaint". Our system could only predict it as "comment".

The bi-LSTMs outperform MLPs as MLPs d not take any positional information into account. We also experimented with POS tags as features but we found that they do not offer any advantage in neural networks, instead they introduce additional noise to the data. The frequent classes exhibit better classification accuracy, as the model can classify them with high confidence. The accuracy was high for Spanish because it had relatively few labels compared to other languages. But the rare classes with low occurrence in the training data are ambiguous and difficult to classify correctly. Japanese was different compared to the rest of the three languages because of its agglutinative nature. Segmentation is a major challenge for Japanese language. So instead of tokenizing words with white spaces, we considered characters as tokens and obtained significant improvements in development and test data. For the improvements in the Japanese system, we used unigram TF-IDF vectors for SVM. No external tools were used for Japanese text segmentation. Our submitted Japanese system used whitespace-separated word vectors for SVM and bi-LSTM.

## 7 Conclusion & Future Work

In this paper, we showed that machine learning approaches and neural networks could achieve comparable accuracy to the systems relying on handcrafted features. The bi-directional LSTMs also performed reasonably well with limited amount of

data.

In the future, we intend to use character embeddings along with the word embeddings to get better representation of a sentence. This will also help in getting a representation for out-of-vocabulary(OOV) words. We can explore multilingual embeddings where words in one language can be mapped to its equivalent in another language. This can also help improve the classification accuracy. We intend to include some linguistic regularization (Qian et al., 2016) while learning the bi-LSTM to take advantage of intensifiers, negative words, positive words and other cue words.

## Acknowledgements

We thank Vandan Mujadia and Pranav Dhakras for their valuable inputs and feedback for us on the paper.

## References

Michael Bentley and Soumya Batra. 2016. Giving voice to office customers: Best practices in how office handles verbatim text feedback. In *Big Data (Big Data), 2016 IEEE International Conference on*, pages 3826–3832. IEEE.

François Chollet et al. 2015. Keras. https://github.com/fchollet/keras.

Corinna Cortes and Vladimir Vapnik. 1995. Support vector machine. *Machine learning*, 20(3):273–297.

Michael Gamon. 2004. Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings of the 20th international conference on Computational Linguistics*, page 841. Association for Computational Linguistics.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5):602–610.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning, ICML-2001*, pages 282–289.

Andrew McCallum, Kamal Nigam, et al. 1998. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Madison, WI.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Qiao Qian, Minlie Huang, and Xiaoyan Zhu. 2016. Linguistically regularized lstms for sentiment classification. *arXiv preprint arXiv:1611.03949*.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Michael Wojatzki, Eugen Ruppert, Sarah Holschneider, Torsten Zesch, and Chris Biemann. 2017. Germeval 2017: Shared task on aspect-based sentiment in social media customer feedback. In *Proceedings of the GSCL GermEval Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*.