Question Classification in Code-Mixed Language

IIT Kanpur , Department of Computer Science and Engineering

Prabuddha Chakraborty (15111027) Archit Rathore (12152)

Introduction

Code-Mixing or Code-Switching is a defined as the embedding of linguistic units such as phrases, words and morphemes of one language into an utterance of another language [source: Wikipedia]. Code-Mixing is prevalent among bi-lingual and multi-lingual individuals.

- “Mein soch raha tha ki we should atleast ek baar unn logo se isko discuss kar lena chahiye.”
- “Annual report mein jo old wala team structure decide kiya tha wahi rehne do.”
- “DOTA ta install kor agee then suvradeep-er team-e dhuke jabi.”

In this project we have tried to tackle the standard NLP question-classification task for code-mixed questions

Motivation

Question classification is the task of finding out the type of the expected answer. This is imperative to solving large number of retrieval and query tasks. It also finds application in Intelligent conversation system, search engine, document retrieval.

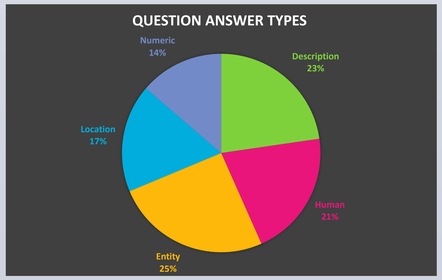
The results of a question classifier can be used to improve upon[Babak Loni: A survey of the Art methods on question classification]:

❖ Reducing the probable answer space
- If a given a question, “Who was the president of India in 2003? ”. The expected answer type would be human, so we only consider the named entities in the candidate answers and in effect reducing the search space.

❖ Choosing Search strategy
- For a question, “What is tectonic plates? ”. As the answer type is ‘Definition’, So our candidate answer would conform to the form: “Tectonic plates are… ” or “Tectonic plate is…”

Question classification for code-mixed questions poses a special problem due to unstandardized variations, alternating semantic models between phrases. To mine voluminous code-mixed content from social media platforms, standard approaches to the problem cannot be applied and hence the need for code-mixed question classifiers arises.

Dataset



There is no standard available dataset of code-mixed questions. Hence we have created our own dataset for the task.

The dataset has the following format-
- CC:FC QuestionText

  CC = Coarse grained Category
  FC = Fine grained Category
  QuestionText = The Code-mixed question

Coarse grained categories are:
- DESC, LOC, ENTY, HUM, NUM

The datset is created based on the English classification dataset by Li and Roth [3, Li rothdatset]. In order to facilitate our conversion process we have made our own transliteration tool. 400 questions were randomly selected and reposed into corresponding hindi-English code-mixed sentences.

Methodology

Approach 2:
- We have collected a 150,000 sentences of code-mixed chat data.
- Normalization of the chat data.
- Train GloVe and Word2Vec models using the refined data.
- Obtain the vector of each question sentence by combining the word vectors provided by the previously trained GloVe and Word2Vec Models

Approach 1:
- Vectorize the question sentences using Bag of N-grams

Train a SVM classifier using the obtained Vectors of the questions.

Algorithm for normalization of the chat data :

procedure NORMALIZE (CODEMIXED_CORPUS C)
  $\forall w \in C$
  if $H_t(w)$!= nil ***then*** return w
  elseif $E(w)$!= nil ***then*** return w
  else return $T(H_h(T^{-1}(w)))$

$H_t \rightarrow Hindi\ Transliterated\ Dictionary$
$H_h \rightarrow Hindi\ Devnagari\ script\ Dictionary$
$E \rightarrow English\ \ Dictionary$
$\mathrm{T} : \mathrm{Hh} \rightarrow H_t$

Word2Vec:

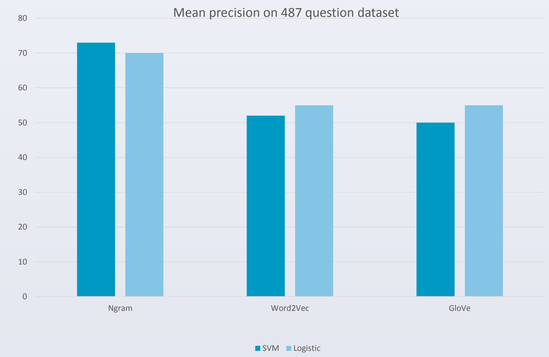
Word2Vec[4] models once trained provides vector representation of words using continuous bag-of-words and skip-gram architecture.

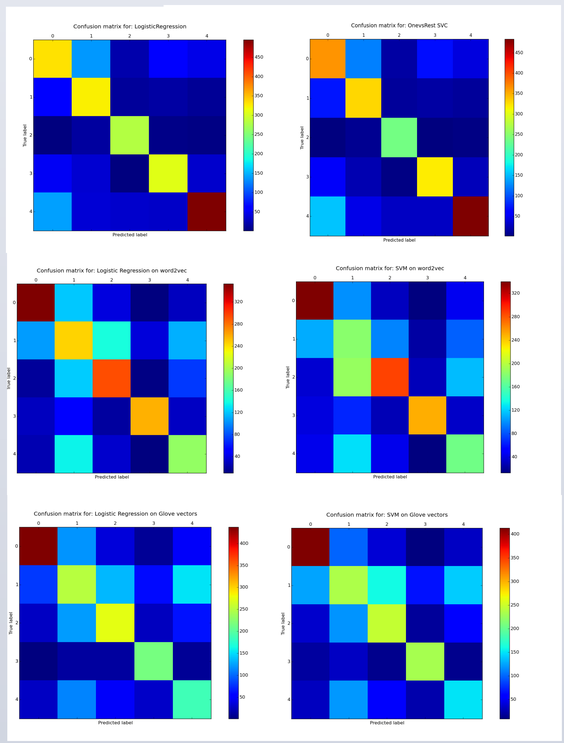
GloVe:

GloVe[3] models gives vector representation of words by utilizing aggregated global word-word co-occurrence statistics.

Results

- Approach 1 – obtained 0.73 mean accuracy using 487 question data with 5% test to train ratio over 100 iterations using SVM on unigram features.
- Approach 2 – obtained 0.546 mean accuracy with same configuration of dataset as above using Logistic regression on Glove vectors.





System	Accuracy (Coarse-Grained)	% Impr. on Baseline
Baseline	55%	-
Baseline + ADJ	57%	2
Baseline + ADJ + Translation + Linear Kernel	62%	7♯
Baseline + ADJ + Translation + RBF Kernel	63%	8♯

Conclusion

- The current result using unigrams surpasses the work of Raghavi et al. [6]
- The word2vec and glove model have decent results with 150,000 lines of data. Larger amounts of data would be able to give better embeddings.
- The classification was done without normalization of the words in the sentences.

Further work

- Will test the effect of normalization on the classification accuracy.
- Use cross validation to tune the classifier parameters.
- Try to generate more code mixed questions for the dataset.

REFERENCES

[1] Xin Li and Dan Roth. Learning question classiers. Proceedings of the 19th international conference on Computational linguistics, 2002.

[2] Xin Li and Dan Roth. Experimental data for question classication. http://cogcomp.cs.illinois.edu/Data/QA/QC/.

[3] Pennington, Jeffrey. 'Glove: Global Vectors For Word Representation'. *Nlp.stanford.edu*. N.p. http://nlp.stanford.edu/projects/glove/.

[4] Code.google.com,. 'Word2vec - Tool For Computing Continuous Distributed Representations Of Words. - Google Project Hosting'. N.p. https://code.google.com/p/word2vec/.

[5] Google Developers,. 'Transliterate (Deprecated) Google Developers'. N.p., 2015. https://developers.google.com/transliterate/.

[6] Khyathi Chandu Raghavi, Manoj Kumar Chinnakotla, and Manish Shrivastava. Answer ka type kya he?: Learning to classify questions in code-mixed language. Proceedingsof the 24th International Conference on World Wide Web, pages 853-858,2015

Acknowledgements

We would like to acknowledge Prof. Amitabha Mukherjee for his guidance and inputs through out the project. We also thank the department and the institute for the resources. Our classmates for their feedback. And Mr. Pranjal Rathore and Ms. Aakriti Mittal for their help in creating the dataset.