iSAI-NLP-AIoT
November 5 - 7
Chiang Mai, Thailand
2022

Workshop on NLP/AI R&D

SENTIMENT POLARITY CLASSIFICATION FOR KHMER

Khim Sokheng, Ye Kyaw Thu, Sethserey Sam

Institute of Digital Research and Innovation (IDRI),
Cambodia Academy of Digital Technology (CADT)

Khim Sokheng and Ye Kyaw Thu contributed equally.

CADT
Cambodia Academy of
Digital Technology

Content

- Introduction
- Related Work
- Corpus Building
- Methodology
- Preprocessing
- Experimental Setup
- Results and Discussion
- Error Rate Analysis
- Conclusion and Future work

Introduction

- ➔ Sentiment polarity classification is a natural language processing (NLP) technique used to determine whether data is positive, negative, or neutral.
- ➔ To the best of our knowledge, there is no freely available polarity corpus for Khmer, and one of the objectives of this research is to develop a human-annotated benchmark corpus for sentiment analysis.
- ➔ This paper presents the first investigation results of the semantic polarity classification on one of the low-resource languages, Khmer.

Introduction (Cont'd)

- ➔ This article studies a well-known shallow neural network classifier named FastText with one of the low-resource languages, Khmer.
- ➔ We show that the FastText model achieved precision and recall of 0.740 even with our small Khmer polarity corpus (9K sentences for training and 1K for testing).

Related Work

➔ ***Rina Buoy et al.*** *(2021)*, Khmer Text Classification Using Word Embedding and Neural Networks.

- Data collection:
  - + Gather data from Wikipedia texts and two main local news website (Thmey Thmey and Sabay)
  - + Dataset is roughly 1 million sentences, around 30 million words
  - + Each news article can have more than one labels
  - + 13,902 articles in total have 4,687 articles with single label
  - + Data split into two smaller datasets multi-class (single label) and multi-label (one or more labels) classification

Related Work (Cont'd)

- Model set up
  - + Word Embedding Model: trained by using SVM and FastText
  - + Classifier Models Setup: trained by linear layer, RNN, and CNN
- Model Training
  - + Multi-class : used cross-entropy loss function
  - + Multi-label : use binary cross-entropy loss function
- Results: RNN is outperform than other model for both multi-class and multi-label

Corpus Building

- ➔ Our Khmer Polarity Corpus is developed by collecting manually sentence that are commonly found on the actual data sources such as local news (SBM News, The Phnom Penh Post, Thmey Thmey, Fresh news), Social media (Facebook, Youtube), Wikipedia, and several website like food, health, sport, tourism (e.g. Healthy Cambodia, Sabay News).
- ➔ This valuable data was gathered around one month to reach 10K sentences and developing the polarity corpus.
- ➔ We analyzed the sentiment words from each sentence and classified these words into three categories (positive, negative and neutral).

Corpus Building (Cont'd)

Khmer Corpus Information

Table 1: Khmer Corpus information

Information	Training	Testing
Number of Sentence	9,015	1,001
Number of Word	698,068	75,893
Frequency of positive	5,251	583
Frequency of negative	2,933	325
Frequency of neutral	830	92

➔ 5,804 positive; 3,258 negative; and 922 neutral (sentences)

Corpus Building (Cont'd)

Khmer Polarity Corpus preparation *(type of sentence)*

➔ In this corpus, the keyword(s) is between three pipes and next to the pipes is a polarity for keyword.

For example,

យើងខ្ញុំនឹងប្តេជ្ញាបន្តផ្តល់នូវផលិតផល ដែលមាន***គុណភាពខ្ពស់***សម្រាប់សុខភាពកូនរបស់អ្នក។
||| គុណភាពខ្ពស់ ||| positive
We are committed to continuing to provide ***high quality*** products for your child's health. ||| high quality ||| positive

Corpus Building (Cont'd)

Khmer Polarity Corpus preparation *(type of sentence)*

- If we found more than one keywords in one sentence, we will keep for all by using delimiter (/).

  For example,

  បន្ទប់នេះ***សំឡេងរំខានណាស់*** ហើយខ្ញុំ***មិនអាចគេងបាន***។
  ||| សំឡេងរំខានណាស់/ខ្ញុំមិនអាចគេងបាន ||| negative
  This room is ***very noisy*** and ***I can not sleep***. ||| very noisy/I can not sleep ||| negative

Corpus Building (Cont'd)

Khmer Polarity Corpus preparation *(type of sentence)*

- If we found the same keyword in one sentence, we will get only one keyword. For example,

  រីឯមូលហេតុ ដែលបង្កជាអគ្គិភ័យនេះឡើង គឺបណ្តាលមកពីរន្ទះបាញ់ តុល្យភាពនៃការ***ខូចខាត***គឺឆេះទីស្នាក់ការចំនួន០១ និង***ខូចខាត***សម្ភារអស់ទាំងស្រុង ។ ||| ខូចខាត ||| negative

  The reason of the fire was caused by lightning, the balance of the ***damage*** was one office fire and complete ***damage*** to equipment. ||| damage ||| negative

Corpus Building (Cont'd)

Khmer Polarity Corpus preparation *(type of sentence)*

- If one sentence have both positive and negative word/phrase, we will decide the main sentiment word/phrase related to overall sentiment in the sentence. For example,

  ពួកខ្ញុំក៏មានការ*ព្រួយបារម្ភ*ដែរសម្រាប់ការអវត្តមានរបស់លឹម ពិសុទ្ធ ប៉ុន្តែពួកខ្ញុំ***ប្រឹងប្រែង***យកលទ្ធផលដើម្បីលឹម ពិសុទ្ធ និងក្រុម"។ ||| ប្រឹងប្រែង ||| positive

  We are also *worried* about Lim Pisoth's absence, but we are ***working hard*** to get the result for Lim Pisoth and his team. ||| working hard ||| positive

Methodology

- ➔ The architecture is similar to Word2Vec but not deeplearning approach (i.e. 1 hidden layer)
- ➔ The words representation are averaged into the sentence representation and directly followed by the output layer
- ➔ This simple architecture works extremely well on classification tasks
- ➔ Comparable to deeplearning approaches

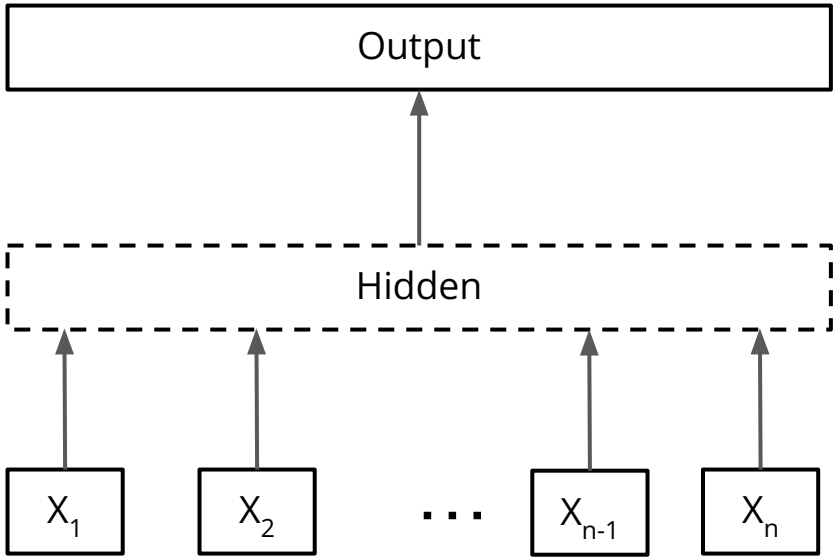
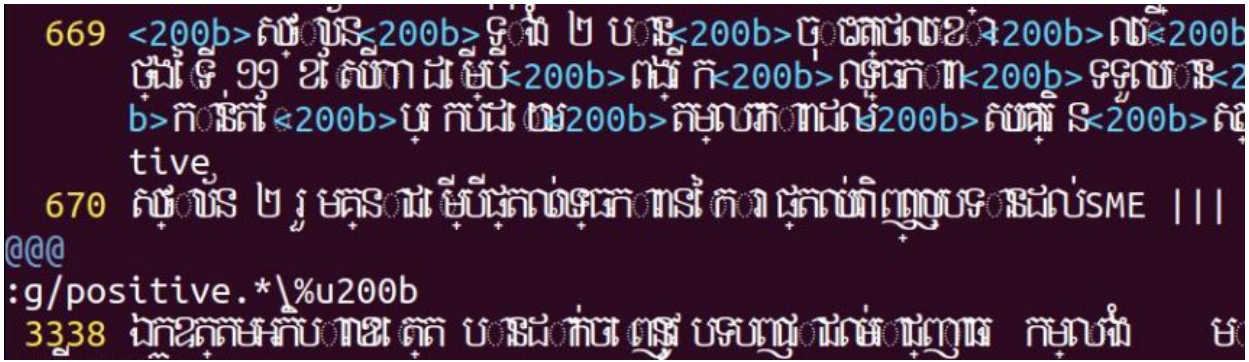


Figure: Model architecture of the FastText t for a sentence with N ngram features $x_1, \ldots, x_n$

Preprocessing

➔ Before we start training and testing, we need to do the data preprocessing on our dataset. In this part, we have checked data which some missing the sentiment keywords or its polarity, pipe (|), spelling checking on polarity, and fixed Khmer zero space character.




```
   1 positiv
5618 positive
   1 •positive
  27 • positive
   1 • ••positive
   3 •• positive
   1 ••• positive
  18 positive •
```

Experimental Setup

- ➔ Traditional machine learning approaches and FastText
- ➔ For the traditional machine learning approaches, including K-nearest Neighbors (KNN), Decision Tree, Random Forest, Support Vector Machine (SVM), Stochastic Gradient Descent (SGD)
- ➔ We selected FastText approach Not only faster training time but also able to train/run on CPU

Results and Discussion

Table 2: Train score and validation score in each model

Model	Unigram Counts	Unigram Tf-Idf	Bigram Counts	Bigram Tf-Idf
KNN	0.64	0.64	0.62	0.6
	0.52	0.52	0.51	0.46
Decision Tree	0.84	0.83	0.83	0.84
	0.52	0.53	0.53	0.53
Random Forest	0.83	0.83	0.84	0.84
	0.58	0.58	0.57	0.57
SVM	0.62	0.6	0.77	0.69
	0.59	0.57	0.56	0.59

Results and Discussion (Cont'd)

Table 2: Train score and validation score in each model

Model	Unigram Counts	Unigram Tf-Idf	Bigram Counts	Bigram Tf-Idf
SGD	0.64	0.63	0.79	0.74
	0.58	0.59	0.57	0.59
SGD Tuning	0.64	0.63	0.79	0.74
	0.59	0.59	0.56	0.58

Results and Discussion (Cont'd)

Table 3: FastText model

Evaluation	**F1-Score**	**Precision**	**Recall**
Positive	0.8152	0.7884	0.8439
Negative	0.6848	0.6746	0.6953
Neutral	0.3308	0.5365	0.2391
P@1	0.74		
R@1	0.74		

Error Rate Analysis

Table 4: Test Result and Error rate in each model

Model	Unigram Counts	Unigram Tf-Idf	Bigram Counts	Bigram Tf-Idf
KNN	0.492	0.533	0.472	0.462
	0.51	0.47	0.53	0.54
Decision Tree	0.538	0.549	0.567	0.524
	0.46	0.45	0.43	0.48
Random Forest	0.535	0.577	0.576	0.579
	0.47	0.42	0.42	0.42
SVM	0.583	0.581	0.577	0.608
	0.42	0.42	0.42	0.39

Error Rate Analysis (Cont'd)

Table 4: Test Result and Error rate in each model

Model	Unigram Counts	Unigram Tf-Idf	Bigram Counts	Bigram Tf-Idf
SGD	0.584	0.588	0.57	0.597
	0.42	0.41	0.43	0.40
SGD Tuning	Best params: {'eta0': 0.009619888350768734, 'learning_rate': 'adaptive', 'loss': 'hinge'} Best score: 0.5926347935337761 Best params: {'alpha': 9.80091439157578e-05, 'penalty': 'l2'} Best score: 0.5921915810558676 Result with the Best SGD Classifier: 0.598 Error Rate: 0.40			

Conclusion and Future work

- ➔ In this paper, we explore several text features and they are unigram-counts, unigram tf-idf, bigram-counts, bigram-tfidf and FastText
- ➔ From the experimental results, FastText achieved the best classification result
- ➔ SVM achieved the highest accuracy among traditional machine learning approaches
- ➔ We plan to extend current corpus and release as publicly available corpus with for further study on Khmer sentiment polarity classification

Reference

[1] Buoy, Rina, Nguonly Taing, and Sovisal Chenda. "Khmer Text Classification Using Word Embedding and Neural Networks." arXiv preprint arXiv:2112.06748 (2021).

[2] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of Tricks for Efficient Text Classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 427–431, Valencia, Spain. Association for Computational Linguistics.


Thank You!