# Understanding Happiness by Using a Crowd-sourced Database with Natural Language Processing

Yixia Wang  Giacomo Spigler

Tilburg University

y.wang_1@tilburguniversity.edu, g.spigler@uvt.nl

## Abstract

In this thesis, there are three studies that delve into a database to know more about Happiness. This crowd-sourced database is called Happy DB, which consists of more than 100,000 descriptions of happy moments collected using Amazon's Mechanical Turk. Today we focus on the first study where we conduct two classification tasks on the said database to learn two critical concepts of happiness, agency and sociality. We apply the state-of-art word embedding algorithm BERT to transform all happy moments to context-sensitive representations and then feed them to a one-layer LSTM. We found that the proposed setup improves performance compared to the previous works.

Full thesis: http://spigler.net/giacomo/files/yixia$_{w}ang_{t}hesis_{2}020.pdf$

## Dataset Description

For these classification tasks, there is already a corpus called CL-Aff corpus[4] where 'sociality' and 'agency' are annotated as an extension to the original happy DB database. **'Sociality'** refers to feeling happy in the presence of others vs alone, while **'agency'** denotes whether the happy moment refers to the participant who reported it or to other people.

### Training Set

| | | Sociality | | |
|---|---|---|---|---|
| | | yes | no | Sum |
| Agency | yes | 3554 | 4242 | 7796 |
| | no | 2071 | 693 | 2764 |
| Sum | | 5625 | 4935 | 10560 |

### Test Set

17,215 Subsamples that contain labels of *sociality* and *agency*.

Demographic information of authors are also given.

## Previous Work

Previous work explored diverse methods involving supervised and semi-supervised learning. Various embedding algorithms were also employed including word2vec [6], GloVe [7], ELMo [8] and word embeddings pre-trained on WikiText-103 corpus[14].

| **Models** | |
|---|---|
| **Supervised Learning** | **Unsupervised Learning** |
| Word Pair Convolutional Model[11] | Autoencoders[1] |
| CNNs [12, 2, 15] | K-means clustering [4] |
| RNN[10] | |
| Bi-LSTM[13] | |
| LSTM[13] | |

## Our Methods

We develop eight baseline models and one-layer LSTM paired with GloVe as a benchmark, to which our main model, one-layer LSTM accompanied by BERT, is compared.

## Main Model: LSTM with BERT

Why LSTM?

1. LSTM addresses problems where other neural networks could not have effectively tackled.
2. It retains the characteristics of varying-length inputs.
3. It includes a short-term component and also a long-term memory component.
4. Since invented, LSTM has documented outstanding performance in tasks such as text generation, text synthesis and sentiment analysis, to name a few.

In order to achieve a better outcome, we trailed several rounds and the hyper-parameter values that contribute to the best performer in both tasks are marked in bold.

Configurations of the LSTM model

| Hyper-parameter | Value |
|---|---|
| Batch Size | **32**, 64, 128, 256 |
| Numbers of Layers | **1**, 2 |
| Units | 32, 48, **60**, 128, 256 |
| Learning Rate | 0.001, 0.003, **0.1** |
| Droupout Rate | **0.2**, 0.4, 0.5 |
| Optimizer | **Adam**, Rmsprop |
| Input Representation | GloVe, **BERT** |

In terms of BERT, it is a transformer-based word embedding algorithm. There are 24 types of BERT models leased online, including BERT-Tiny, BERT-Small, etc. We use BERT-Base model(Uncased: 12-layer Transformer, 768 dimensions, 12-heads, 110M parameters)[3]. **Max sequence length** is set as 128. **Padding** and **truncating** skills are involved in fixing the length of each description to 128. Therefore, for each input text, BERT outputs a tensor of shape (128, 768) with one vector per token. Out of 12 layers, We summed the last four layers as a pooling strategy to obtain a fixed representation for each happy moment description.

## Baseline Models

Eight baseline models are developed by applying traditional machine learning algorithms, including Support Vector Machine(SVM), Random Forest, Logistic Regression(Log Reg) and Naive Bayes. Each of them is implemented with two sets of word embedding algorithms and they are: Bag of Words(BOW) and Bag of Words with a TF - IDF transformation(BOW tf-idf).

On top of these machine learning approaches, LSTM paired with GloVe is also used as a benchmark. GloVe is an unsupervised learning algorithm for distributed word representation, whose performance we want to compare to that of BERT.

## Results

The results presented here suggest that the proposed model with BERT outstrips the other two in both tasks and in all evaluation methods. It achieves 86.42% accuracy, 90.42% F1 score and 91.41% area under curve(auc) in the agency classification task. As to the sociality task, as the samples are more balanced, the results are better. Our model with BERT hit 97.11% in auc. Its F1 score is also the highest, boasting 93.49%. The model also achieved 93% in accuracy, outperforming the previous best model(EMLo+LSTM).

Result of Main Model

| | Sociality | | | Agency | | |
|---|---|---|---|---|---|---|
| Models | Accuracy | F1 Score | AUC | Accuracy | F1 Score | AUC |
| ELMo + LSTM(publication) | 92.00% | 93.00% | None | 85.00% | 90.00% | None |
| GloVe + LSTM | 90.14% | 90.89% | 95.70% | 83.70% | 88.55% | 89.41% |
| BERT + LSTM | **93.00%** | **93.49%** | **97.11%** | **86.42%** | **90.42%** | **91.41%** |

**Table 1.** Accuracy of the proposed LSTM+BERT model on Agency and Sociality classification.

As to baseline models, the linear SVM with BOW as the word representation method performs well on the first task, achieving a score of 90.49%. Logistic Regression outperforms others, reaching 80.65% of accuracy in agency classification.

Result of Baseline Models

| | Accuracy | |
|---|---|---|
| Models | Sociality | Agency |
| Logistic Regression(BOW-tfidf) | 88.90 | 78.11 |
| Logistic Regression(BOW) | 89.77 | **80.65** |
| Linear SVM(BOW-tfidf) | **90.49** | 78.34 |
| Linear SVM(BOW) | 90.01 | 76.51 |
| Random Forest(BOW-tfidf) | 56.95 | 70.61 |
| Random Forest(BOW) | 56.91 | 70.60 |
| Naive Bayes(BOW-tfidf) | 70.80 | 46.11 |
| Naive Bayes(BOW) | 67.90 | 45.78 |

Table 5: Best performing baseline models on Sociality and Agency classification respectively.

## Reference

- [1] Byung-Chull Bae, Yun-Gyung Cheong, and Youngrok Song. "Modeling Happiness Using One-Class Autoencoder". In: (2019).
- [2] Daniel Claeser. "Affective content classification using convolutional neural networks". In: AffCon@ AAAI. 2019.
- [3] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: arXiv preprint arXiv:1810.04805 (2018).
- [4] Kokil Jaidka et al. "The CL-Aff happiness shared task: Results and key insights". In: (2019).
- [5] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: arXiv preprint arXiv:1412.6980 (2014).
- [6] Tomas Mikolov et al. "Distributed representations of words and phrases and their compositionality". In: Advances in neural information processing systems. 2013, pp. 3111–3119.
- [7] Jeffrey Pennington, Richard Socher, and Christopher D Manning. "Glove: Global vectors for word representation". In: Proceedings of the 2014 con- ference on empirical methods in natural language processing (EMNLP). 2014, pp. 1532–1543.
- [8] Matthew E Peters et al. "Deep contextualized word representations". In: arXiv preprint arXiv:1802.05365 (2018).
- [9] Lutz Prechelt. "Early stopping-but when?" In: Neural Networks: Tricks of the trade. Springer, 1998, pp. 55–69.
- [10] Arun Rajendran, Chiyu Zhang, and Muhammad Abdul-Mageed. "Happy together: Learning and understanding appraisal from natural language". In: arXiv preprint arXiv:1906.03677 (2019).
- [11] Michael Saxon et al. "Word pair convolutional model for happy moment classification". In: (2019).
- [12] Baohua Sun et al. "Squared english word: A method of generating glyph to use super characters for sentiment analysis". In: arXiv preprint arXiv:1902.02160 (2019).
- [13] Bakhtiyar Syed et al. "Ingredients for happiness: Modeling constructs via semi-supervised content driven inductive transfer learning". In: Proceed- ings of the 2nd Workshop on Affective Content Analysis@ AAAI (AffCon2019), Honolulu, Hawaii (January 2019). 2019.
- [14] Johnny Torres and Carmen Vaca. "CL-Aff Deep semisupervised cluster- ing". In: (2019).
- [15] Weizhao Xin and Diana Inkpen. "Happiness Ingredients Detection using Multi-Task Deep Learning". In: (2019).