CS839 Stage 1 Report: Information extraction from natural text

Xiuyuan He [*1] **Chenlai Shi** [*1] **Mingren Shen** [*1]

1. Name of all team members

- Xiuyuan He
- Chrissie Watts
- Mingren Shen

2. Entity Type

We want to extract **people names** from moive review texts. The moview reviews are from Large Movie Review Dataset v1.0 (Maas et al., 2011) by Stanford University [1].

Examples are:

- Gina Yashere
- Chrissie Watts
- John's

Detailed rules of the entity type are:

1. Prefix and Titles like Mr., Mrs., Ms., Director, etc are **not included**
2. Suffix Names like Sr., Jr., IV, etc **are included**
3. Names form a possessive with the suffix -'s like John's, Mike's **are included**
4. Both Actor Names and Movie Character Names **are considered names**
5. People Names used in Movie Titles like "Mr. & Mrs. Smith" or Company Names like "Warner Bros. Entertainment Inc" **are considered names**

We use "<>" and "</>" to mark up all the occurrences of person names. So for the example above, we will mark them like this:

- <> Gina Yashere </>
- <> Chrissie Watts </>
- <> John's </>

3. Data Set

3.1. the total number of mentions that you have marked up

There are **1695** mentions of person names are marked up.

3.2. the number of documents in set I, the number of mentions in set I

There are **200** documents in set I and **1103** mentions of person names are marked up.

3.3. the number of documents in set J, the number of mentions in set J

There are **100** documents in set J and **592** mentions of person names are marked up.

4. Pre-processing

For the marked up text files, we do the following steps to clean the generated examples.

- delete all numbers
- delete all punctuation
- delete all stopping words
- delete all 4 words examples(we do not see this in our data set)

5. Training and Model Selection

5.1.

We chose 5 different machine learning models:

- SVM with RBF kernel
- Decision Tree using CART Algorithm
- Random Forest
- Logistic Regression
- Linear Regression

[*]Equal contribution [1]University of Wisconsin,Madison , USA. Correspondence to: Xiuyuan He <xhe75@wisc.edu>, Chenlai Shi <cshi29@wisc.edu>, Mingren Shen <mshen32@wisc.edu>.

CS 839 Class Project, UW-Madison, 2018. Copyright 2018 by the author(s).

[1] http://ai.stanford.edu/~amaas/data/sentiment/

5.2.

5.3. Any Rule-based Post-processing?

We are using one main rules for post-processing of the prediction results.

Software and Data

We provide all our data and program in Github and you can check them online https://github.com/iphyer/CS839ClassProject.

We use scikit-learn (Pedregosa et al., 2011) as our machine learning program library and Pandas (McKinney, 2015) for data processing.

References

Maas, Andrew L., Daly, Raymond E., Pham, Peter T., Huang, Dan, Ng, Andrew Y., and Potts, Christopher. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P11-1015.

McKinney, Wes. pandas: a python data analysis library. *see http://pandas. pydata. org*, 2015.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.