Analyzing Well-Formedness of Syllables in Japanese Sign Language

Satoshi Yawata* **Makoto Miwa** **Yutaka Sasaki** **Daisuke Hara**
Toyota Technological Institute
2-12-1 Hisakata, Tempaku-ku, Nagoya, Aichi, 468-8511, Japan
yawata@nsk.com
{makoto-miwa,yutaka.sasaki,daisuke.hara}@toyota-ti.ac.jp

Abstract

This paper tackles a problem of analyzing the well-formedness of syllables in Japanese Sign Language (JSL). We formulate the problem as a classification problem that classifies syllables into well-formed or ill-formed. We build a data set that contains hand-coded syllables and their well-formedness. We define a fine-grained feature set based on the hand-coded syllables and train a logistic regression classifier on labeled syllables, expecting to find the discriminative features from the trained classifier. We also perform pseudo active learning to investigate the applicability of active learning in analyzing syllables. In the experiments, the best classifier with our combinatorial features achieved the accuracy of 87.0%. The pseudo active learning is also shown to be effective showing that it could reduce about 84% of training instances to achieve the accuracy of 82.0% when compared to the model without active learning.

1 Introduction

Japanese Sign Language (JSL) is a widely-used natural language different from Japanese. JSL vocabulary needs to be expanded because JSL vocabulary seems much smaller than Japanese one (Tokuda and Okumura, 1998) and JSL words for new concepts are always required (Japanese Federation of the Deaf, 2011). Many JSL words and syllables, which are basic units that compose words, are newly coined to meet these requirements, e.g., (Japanese Federation of the Deaf, 2011). However, some of the syllables are *ill-formed*, or unnatural for JSL natives, since these new syllables are often coined by non-natives (Hara, 2016a). This ill-formedness is problematic since this can cause miscommunication and also erroneous learning for JSL non-natives. Figure 1 illustrates the examples of well-formed and ill-formed JSL syllables (monosyllable words): "basis"[1] and "avocado"[2].

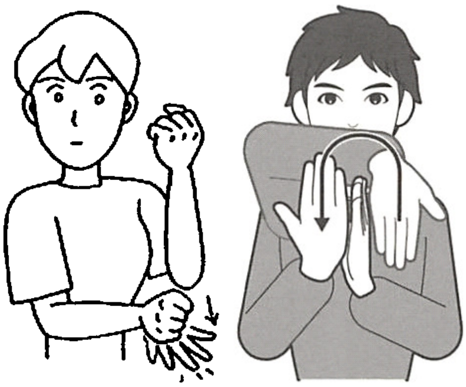

Figure 1: Examples of well-formed (left) and ill-formed (right) JSL syllables. They are also monosyllable words: the left syllable means "basis" and the right syllable means "avocado" (Yonekawa, 1997; Japanese Federation of the Deaf, 2011).

The phonology and phonotactics of JSL have not been well studied and the causes for this ill-formedness have not been revealed. Natives can distinguish such syllables, but they cannot clearly explain the causes since the ill-formedness stems from their intuition. It is thus difficult to distinguish ill-formed syllables from well-formed ones without the help of natives. A practical approach is required to analyze and understand the ill-formedness of syllables objectively to exclude

[*] Currently at NSK Ltd.

[1] Stand up the left elbow, touch the closed right hand and open it downwards.

[2] Put the right little finger to the back of the left hand standing up and move the right hand to cut it towards the palm of the left hand

Proceedings of the The 8th International Joint Conference on Natural Language Processing, pages 26–30,
Taipei, Taiwan, November 27 – December 1, 2017 ©2017 AFNLP

the ill-formed syllables and avoid producing them with little burden on native signers.

In this paper, we describe an approach to model the well-formedness of syllables in JSL as a classification problem and analyze the cause of the well-formedness. We build a data set that contains 2,891 hand-coded syllables with their well-formedness. Based on the data set, we train an L1-regularized logistic regression classifier using a fine-grained feature set to investigate the applicability of machine learning (ML) approaches and to find the differences between well-formed and ill-formed syllables. We also apply pseudo active learning (Settles, 2009) to the data to investigate the possibility in reducing the annotation costs.

As far as we know, this is the first approach that tackles the well-formedness of JSL syllables with ML. We got the following insights from our experiments. First, the syllables can be classified into well-formed or not in the accuracy of 87.0% with the simple classifier on sparse fine-grained features. Second, we disclosed features that are useful for the classification. Third, we show that active learning can reduce the annotation costs. We will make the annotated data available upon request[3].

2 Method

This section explains how we define and tackle the classification problem to analyze the well-formedness of JSL syllables. We first define the representation of syllables. We then explain the classification and pseudo active learning methods.

2.1 Syllable representation

JSL is a visual language, and the syllables are expressed visually. To avoid the difficulty in dealing with the visual language[4], we decide to hand-code syllables. JSL syllables are usually composed of three elements: handshapes, movements, and locations (Kimira et al., 2011).

We hand-code JSL syllables with the encoding scheme by Hara (2016b), which is extended from Hara (2003). Each syllable is represented with seven components in this coding: `types`, `handshapes`, `locations`, `movements`, `contacts`, `directions of palms`, and `directions of wrists`. We here briefly explain these components: `Types` denote the number of hands used and, if two hands are involved, the information about whether both hands have the identical or different handshapes, and whether both hands move together or not. `Handshapes` represent the handshape types. `Locations` correspond to 28 locations of hands on or around the body such as the eye, the shoulder, neutral space, i.e., space in front of the signer, and so on. `Movements` are the movement types of hands such as path movement, orientation change movement, and handsape change movement, and their relationships such as synchronous movement and alternating movement. `Contacts` indicate whether and when both hands have contact in the syllable execution. `Directions of palms` show which direction the palm faces. `Directions of wrists` denote directions to which the tip of the metacarpal bones point.

Syllables have little overlap in this coding and it is impossible to find the discriminative characteristics between well-formed syllables and ill-formed ones, so we decompose the components in the coding into a set of fine-grained binary features, aiming that the features are shared among syllables without losing the original information. `Types` are represented with nine binary features, e.g., whether both hands are used, whether both hand movements are symmetric, etc. `Handshapes` are decomposed into 208 binary features to represent whether each finger in hands is used and whether each finger joint in hands is stretched, loosely bent, or bent. Similarly, we define 98 binary features for `locations`, 398 for `movements`, 171 for `contacts`, and 62 for `directions of palms`, and 62 for `directions of wrists`. With this decomposition, we define 1,017 binary features in total.

2.2 Well-formedness classification

We employ an L1-regularized logistic regression classifier to classify well-formed and ill-formed syllables. Training instances are not so many and it is unknown how ML approaches work on this problem, so we decide to employ this simple classifier as the first step toward this problem. We use the L1 penalty to encourage the model to be sparse, expecting that we can make the finding of discriminative features easier. We also consider adding the **combinatorial features** of two binary

[3] Please contact the last author for data related inquiry.

[4] We left the automatic coding of visually-expressed syllables as future work.

	Accuracy	F1
most frequent	0.826	–
binary features	0.837	0.533
+ combinatorial features	0.870	0.613

Table 1: Classification results

features so that we can get more descriptive features.

2.3 Pseudo active learning

There are plenty of JSL syllables in practice, and it is infeasible to manually annotate these syllables[5]. We apply pseudo active learning to the data set and investigate the possibility of reducing the annotation cost. We employ two strategies: an **uncertainty** sampling strategy that chooses the least confident instances (Lewis and Catlett, 1994) and a **certainty**-based strategy that chooses most negative (ill-formed) instances, which was shown to be effective for imbalanced data sets (Fu and Lee, 2013; Miwa et al., 2014).

3 Evaluation

3.1 Experimental settings

Data sets: We employed 25 JSL natives to hand-code 2,891 syllables and annotate their well-formedness. The syllables are taken from Yonekawa (1997) and the book series of "Our Sign Language", e.g., (Japanese Federation of the Deaf, 2011). We split the syllables into training and test data sets. The training data set contained 2,053 well-formed (positive) syllables and 538 ill-formed (negative) syllables. The test data set contained 238 positive and 52 negative syllables.

Well-formedness classification: We employed the L1-regularized logistic regression classifier in scikit-learn[6]. We evaluated the classification performance by using both the classification accuracy and F1 score on negative, ill-formed syllables as the evaluation metrics. We also compared two models to check whether the combinatorial features help: one uses binary features and the other uses combinatorial features of two binary features. We tuned the regularization parameter by a 20-fold cross validation (CV) on the training data.

Pseudo active learning: Using the classification accuracy as the evaluation metric, we compared three models: random baseline with binary features (random), active learning with binary features (active), and active learning with binary and combinatorial features (active(combi)). We also compared the two active learning strategies using binary and combinatorial features. We built the initial classifier by training the classifier on 20 instances consisting of 10 well-formed and 10 ill-formed syllables. We added labeled instances one by one in active learning. We tuned the regularization parameter using the 20-fold CV each time 50 instances are added by active learning.

3.2 Results

We first examined the number of features that appeared in the data set. For binary features, 849 out of 1,017 features appeared in the data set. This shows there are some features that rarely or never appear in JSL syllables. Similarly, not all combinatorial features appeared in the data set, and 174,986 out of 359,976 (*i.e.* $\binom{849}{2}$=849×848/2) features appeared. This is mainly because some binary features are disjunctive and their combinations are physically impossible.

Next, we evaluated the classification performance on the test data set (Table 1). Our classifiers produced better accuracy than did the most frequent baseline that always predicted syllables as well-formed. These high accuracies show that our classifiers can detect relatively few ill-formed syllables. The F1 scores are still low, which indicates that we need to investigate how to alleviate the data imbalance problem. This table also shows that the combinatorial features are useful for improving the performance.

Table 2 lists up some contributing features in the model. Among the top 20 features, 9 and 11 features were related to dominant and non-dominant features respectively for binary features, whereas 7, 2, and 11 features were related to dominant, non-dominant, and both hands respectively for combinatorial features. These differences and the performance difference between the features indicate that the relation of both hands are important to decide the ill-formedness.

Figure 2 shows the learning curves of three models (random baseline, active learning, active learning with combinatorial features) explained in Section 3.1 during **active learning**. Each curve in this figure shows the average of 10 runs. This shows active learning work well compared to

[5] We need an established way to automatically code JSL syllables beforehand, e.g., by extending Sako et al. (2016).

[6] http://scikit-learn.org

Dominant hand
Second joints of middle and ring fingers are bent The base of ring finger is bent and the palm direction is diagonally forward The hand moves according to an orbital movement and the palm direction is backward.
Both hands
Movements are not symmetric and there is no contact at the end of a syllable Different handshapes and the direction of the metacarpal bone of the dominant hand is upward Symmetric handshapes and no contact at the beginning of a syllable

Table 2: Examples of contributing combinatorial features

the random baseline. From this figure, random baseline required 1,284 instances while the active learning 184 to achieve 82% in accuracy, so the active learning need about 14.3% of the training data compared to the random baseline. The use of combinatorial features produces slightly worse results, but the final performance is higher than one without combinatorial features which indicates that the combinatorial features work well only with enough training instances.

Lastly, we compare the two active learning strategies in Figure 3. Certainty-based method worked slightly worse than uncertainty-based method did, but the difference is small and, as a whole, both strategies work almost similarly. This result is interesting since the certainty-based strategy focuses only on ill-formed syllables.

4 Related work

Although automatic sign language analysis has been widely studied since 1990s (Starner et al., 1998; Ong et al., 2005), there are relatively few studies on computational approaches to JSL.

Kimira et al. (2011) proposed a JSL dictionary consisting of over 2,000 JSL sign[7]. Each sign is defined with handshapes, motions, and locations, and a movie is attached to the sign. They did not deal with the well-formedness of JSL syllables.

Studies on automatic recognition of JSL are also relatively few, and most of them aim at a small number of syllables or signs. Sako et al. (2016) recently proposed automatic JSL recognition using Kinect v2. They used contour-based handshape recognition, and they recognized hand location and motion by Hidden Markov Models and Gaussian Mixture Models. They evaluated their system on 223 JSL signs. The combination of our method with these automatic recognition methods is one of the interesting research directions.

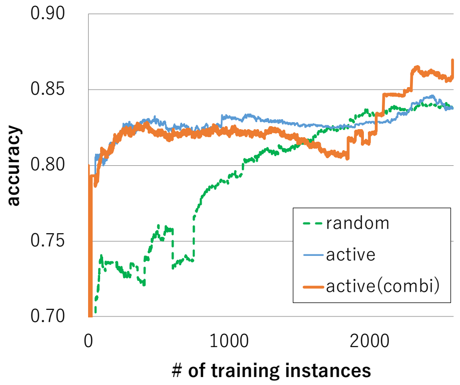

Figure 2: Learning curve with pseudo active learning

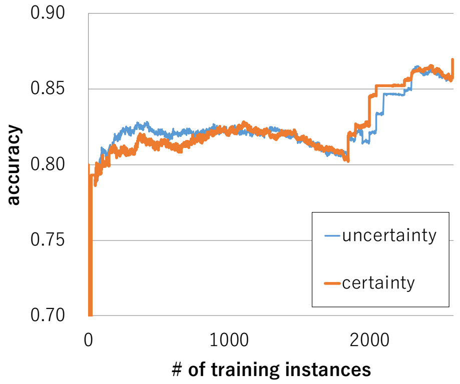

Figure 3: Comparison of pseudo active learning strategies

5 Conclusion

This paper tackled a problem of analyzing the well-formedness of JSL syllables. We created the data set consisting of 2,891 hand-coded syllables with their well-formedness. We then built and evaluated classifiers using the fine-grained binary features on the classification of syllables into well-formed or not. We also investigated the possibility of active learning on the analysis of the well-formedness. The results show that our classifier achieves 87.0% in accuracy and that the active learning can reduce the number of annotations.

[7] A sign consists of one or more syllables

As future work, we would like to incorporate more sophisticated ML approaches such as kernels and deep neural networks to consider more combinations of features. We also would like to develop a system that can code visual syllables into our features to make our method practical to support defining new syllables.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers JP16H03813 and JP15K02536 and DAIKO FOUNDATION.

References

JuiHsi Fu and SingLing Lee. 2013. Certainty-based active learning for sampling imbalanced datasets. *Neurocomputing*, 119:350–358.

Daisuke Hara. 2003. *A Complexity-Based Approach to the Syllable Formation in Sign Language*. Ph.D. thesis, The University of Chicago, Chicago, IL.

Daisuke Hara. 2016a. An information-based approach to the syllable formation of Japanese Sign Language. In Masahiko Minami, editor, *Handbook of Japanese Applied Linguistics*, chapter 18, pages 452–482. Gruyter Mouton, Boston, MA.

Daisuke Hara. 2016b. *New Coding Manual for Japanese Sign language*. (In Japanese).

Japanese Federation of the Deaf, editor. 2011. *Our sign language 2011: new sign language*. Japanese Federation of the Deaf, Tokyo, Japan. (In Japanese).

Tsutomu Kimira, Daisuke Hara, Kazuyuki Kanda, and Kazunari Morimoto. 2011. Expansion of the system of jsl-japanese electronic dictionary: An evaluation for the compound research system. In *Proceedings of the 2nd International Conference on Human Centered Design*, HCD'11, pages 407–416, Berlin, Heidelberg. Springer-Verlag.

David D. Lewis and Jason Catlett. 1994. Heterogeneous uncertainty sampling for supervised learning. In *Proceedings of the Eleventh International Conference on Machine Learning*, pages 148–156. Morgan Kaufmann.

Makoto Miwa, James Thomas, Alison OMara-Eves, and Sophia Ananiadou. 2014. Reducing systematic review workload through certainty-based screening. *Journal of biomedical informatics*, 51:242–253.

Sylvie CW Ong, Surendra Ranganath, et al. 2005. Automatic sign language analysis: A survey and the future beyond lexical meaning. *IEEE transactions on pattern analysis and machine intelligence*, 27(6):873–891.

Shinji Sako, Mika Hatano, and Tadashi Kitamura. 2016. *Real-Time Japanese Sign Language Recognition Based on Three Phonological Elements of Sign*. Springer International Publishing, Cham.

Burr Settles. 2009. Active learning literature survey. Computer Sciences Technical Report 1648.

Thad Starner, Joshua Weaver, and Alex Pentland. 1998. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1371–1375.

Masaaki Tokuda and Manabu Okumura. 1998. Towards automatic translation from japanese into japanese sign language. *Assistive Technology and Artificial Intelligence*, pages 97–108.

Akihiko Yonekawa. 1997. *Japanese – Japanese Sign Language Dictionary*. Japanese Federation of the Deaf, Tokyo, Japan. (In Japanese).