Full Sentence Verb Identification and Conjugation Classification in Japanese with a Bidirectional Long Short-Term Memory Model

Theodore Chen
tchen25@bu.edu

Abstract

This paper highlights the problems and limitations of computationally classifying Japanese verbs through a rule-based approach, along with the feasibility of using Recurrent Neural Networks (RNN) like Long Short-Term Memory (LSTM) models to classify Japanese verbs given the context of an entire sentence.

Code for this project can be accessed here.

1 Introduction

Through the creation of a reliable way to label Japanese verbs based on the sentences they are found in, one can generate large datasets of novel sentences as practice material for understanding complex sentences.

1.1 Japanese Verbs

Japanese verbs, unlike English verbs, pack a significant amount of information inside its structure without much use of outside auxiliary words. This makes conjugating verbs in Japanese incredibly simple, as in order to express a certain tense, voice, or level of politeness, one only needs to remember a set of rules that has minimal exceptions.

However, the opposite is not true. In order to identify and distinguish between certain types of verbs, an entire sentence needs to be accurately understood. Two completely different conjugations can appear identical, and without knowing the context of the sentence or prior statements, it cannot be identified.

1.2 Approach

This means that while it is easy to use hard-coded rules to conjugate a Japanese verb, a method that also takes into account an entire sentence is necessary to accurately distinguish a verb's conjugation.

To establish a baseline, I will use a multi-class logistic regression classifier with character level n-grams as features (which will simulate the ability to classify a verb by only observing the verb and minimal/no context). Then, I will compare this model with a bidirectional LSTM model that is able to account for an entire sentence's context (which will simulate the ability to classify a verb by observing the entire sentence).

2 Background

2.1 Japanese Grammar Overview

Japanese grammar differs significantly from English, both structurally and morphologically. Unlike English, which has the grammatical word order SVO (Subject-Verb-Object), Japanese's word order is SOV (Subject-Object-Verb). Additionally, Japanese marks nouns with particles in order to indicate their role in the sentence (e.g., subject, object, topic, location), rather than word order like in English.

In terms of verb conjugation, Japanese's morphology can be classified as agglutinative, meaning verbs and adjectives are built from a root that is adjusted with different suffixes appended to it to express tense, voice, and politeness. This differs from English where these traits are expressed through a combination of word order, word choice, the use of auxiliary verbs, and morphological adjustments.

Due to this difference, there exist verb conjugations in Japanese specifically for politeness and voice.

2.2 Morpheme Change Examples

Normal Tone, Active, Present

学生	が	宿題	をする
gakusei	ga	shukudai	wo suru

"The student does the homework. (normal)"

Normal Tone, Active, Past

学生	が	宿題	をした
gakusei	ga	shukudai	wo shita

"The student did the homework. (normal)"

Polite Tone, Active, Past

学生	が	宿題	をしました
gakusei	ga	shukudai	wo shimashita

"The student did the homework. (polite)"

Normal Tone, Passive, Past

先生	が	学生	に	宿題をさせた
sensei	ga	gakusei	ni	shukudai wo saseta

"The teacher made the student do the homework. (normal)"

Polite Tone, Passive, Past

先生	が	学生	に	宿題をさせました
sensei	ga	gakusei	ni	shukudai wo sasemashita

"The teacher made the student do the homework. (polite)"

2.3 Tense

In the examples above, the English and Japanese verbs both have their tense indicated by adjustments to the morphology of the word itself. In 2.2.1 and 2.2.2, 「する」(suru) becomes「した」(shita) for the past tense, and "does" becomes "did".

2.4 Politeness

In Japanese, politeness is also embedded into the suffix: 「ます」(masu) for present tense and 「ました」(mashita) for past tense. In 2.2.2 and 2.2.3, the English sentences are identical regardless of the level of politeness indicated by the Japanese sentences.

2.5 Voice

In Japanese, voice is also indicated through the suffix, where in 2.2.2 and 2.2.4: 「した」(shita) is for past tense and active voice, while 「させた」(saseta) is for past tense and passive voice. In English, voice is indicated through word order

2.6 Non-Morpheme Embedded Examples

2.6.1 Normal Tone, Active Potential, Past

お姉さん	は	りんご	が	食べられた
oneesan	wa	ringo	ga	taberareta

"My sister was able to eat the apple. (normal)"

Normal Tone, Passive, Past

お姉さん	に	りんご	を	食べられた
oneesan	ni	ringo	wo	taberareta

"The apple was eaten by my sister. (normal)"

2.7 Non-Morpheme Embedded Semantics

In 2.6.1 and 2.6.2, the verbs in the Japanese sentences are identical, both being 「食べられた」(taberareta), which could mean "to be eaten", or "can eat". Classifying the difference between these two verbs would be impossible without looking at the rest of the sentence, as the context is embedded in the particles attached to the nouns 「お姉さん」(oneesan: "sister") and 「りんご」(ringo: "apple").

In 2.6.1, "sister" is marked by 「は」(wa), indicating the topic of the sentence, and "apple" is marked with「が」(ga), indicating the subject of the sentence. In 2.6.2, "sister" is marked by 「に」(ni), indicating the passive agent, and "apple" is marked by 「を」(wo), indicating the direct object.

These 4 particles also play other roles in other types of sentences varying based on context, making it difficult for Japanese learners to parse their application, and ultimately the meaning of a sentence quickly.

3 Method

3.1 Dataset

There is a lack of datasets that are labeled with the conjugation type of Japanese verbs, making it necessary to automate a way to label prior datasets with a set of established rules. There is also a lack of existing Japanese verb classifier projects.

One dataset was labeled and used to train the model: the Tanaka Corpus (Project, 2021).

3.1.1 Labeling Process

The process taken to identify the verbs in a sentence, then label them according to their conjugation type is as follows:

1. Use the spaCy library (Honnibal et al., 2020) (which utilizes mecab (Kudo, 2005)) to tokenize a sentence into individual morphemes. In this process, POS (parts-of-speech) including verbs are identified then lemmatized (their root is extracted), their conjugation rule category is determined, and the presence of certain relevant particles are saved.

2. Using the lemmatized form of a verb and its conjugation rule category, find all possible conjugations for the verb and add it to a dictionary.

3. Using this list of conjugations, determine what conjugation type the original form of the verb from the sentence is using, labeling the conjugation type (e.g. potential, causative-past)

4. Add identified verbs for each sentence and their corresponding conjugation types to the data. Conjugations are formatted based on the mecab tagset (Sketch Engine, n.d.)

See Appendix A for a flow chart of the process.

3.1.2 spaCy Parsing and Tokenization

The spaCy library uses a BERT transformer model to identify individual morphemes in a sentence as tokens, lemmatizing where applicable, identifying conjugation rule category, and tagging it with it's part-of-speech.

Using this information, verbs are identified based on if it was marked as such in the POS-tagging process.

In order to conjugate a Japanese verb, it's lemmatized (root/dictionary) form and conjugation rule are required. There are 3 main categories of conjugation rule, which (mostly) uniformly conjugate a Japanese verb from it's a certain category the same way. See Appendix B for an overview of these categories, and how they are categorized.

The other classification method is the use of 「に」(ni), 「を」(wo), and「が」(ga) for distinguishing the potential and passive form.[1]

For example, the verb for "to eat" and "to see" are conjugated the same way:

[1]This method is not robust as implemented due to there being additional exceptions, and the semantic meaning of the nouns being marked need to be analyzed to increase it's accuracy, more in the conclusion

「食べる」	→	「食べた」
(tabe-ru)	→	(tabe-ta)
「見る」	→	「見た」
(mi-ru)	→	(mi-ta)

Where the ending in dictionary form 「る」 (ru) is dropped and changed to 「た」 (ta) for the past tense.

Finally, particle data is also stored in the output to aid in labeling the previous ambiguous case in 2.6, where the passive and potential conjugations of one verb category appear identical.

3.1.3 Conjugation

Conjugation rules were applied using a modified script (Fonseca, 2021a), with lemmatization verification being done with jamdict (Fonseca, 2021b).

3.1.4 Cleaning

The dataset was cleaned by removing sentences where no verbs were identified or identified ones couldn't be conjugated. Additionally, sentence segments were identified, where each segment was split by a new identified verb.

Sentences that were removed included sentences in the event the labeling has failed, see conclusion for more.

Libraries used to accomplish this include numpy (Harris et al., 2020) and pandas (pandas development team, 2020).

The Tanaka Corpus (Project, 2021) had 147,865 sentences, where 107,981 sentences remained, with 171,354 verbs/sentence segments.

The following verb types were also removed due to there being one or less example in the dataset: potential-negative-provisional-conditional, causative-passive-negative, causative-negative-nominal, causative-past-polite-conditional, causative-passive-colloquial-past, causative-negative-polite, passive-past-polite-conditional, causative-wish-past, passive-wish-past

3.2 Models

3.2.1 Logistic Regression Model

The input features of this model are character level n-grams ($n = 1 - 3$) for each sentence segment, and the output is a sentence divided into segments by verb, along with the conjugation type of that verb.

The model is implemented in Python (van Rossum and the Python Development Team, 2009), using the sklearn library (Pedregosa et al., 2011) for its LogisticRegression class, CountVectorizer (featurization), training/test split, and classification report.

This model is meant to simulate a verb-only approach to classifying verbs, where only the characters of the verb itself are observed in order to classify a verb. It's expected that this model will perform poorly, especially on potential and passive form where context is required for correct classification.

3.2.2 Bi-directional LSTM Model

The input features for this model are character level embeddings for the entire sentence, and the output is identical to the baseline.

The model is implemented in Python (van Rossum and the Python Development Team, 2009), using the sklearn library (Pedregosa et al., 2011)) for its training/test split and classification report, as well as the PyTorch library (Paszke et al., 2019) for its dataset/dataloader, and BiLSTM class.

This model is meant to simulate a full sentence approach to classifying verbs, where the entire sentence is observed for context on how to classify a verb. It's expected that this model will be significantly better than the logistic regression model, especially on potential and passive form where context is required for correct classification.

4 Results

As expected, the performance of the baseline was lower for all classes. The macro F1 values of the causative, potential, and passive verb class groups in Table 1 and 2. Additionally, classes that were previously unclassified were able to be classified in Figures 1 and 2.

Table 1: Baseline Group Analysis

Group	Classes	Support	Macro F1	Weighted F1
Causative	19	301	0.643	0.849
Potential	9	36	0.464	0.595
Passive	21	1173	0.590	0.933
Other	24	32770	0.915	0.983

Table 2: BiLSTM Group Analysis

Group	Classes	Support	Macro F1	Weighted F1
Causative	19	301	0.990	0.987
Potential	9	36	0.926	0.906
Passive	21	1173	0.900	0.983
Other	24	32770	0.993	0.999

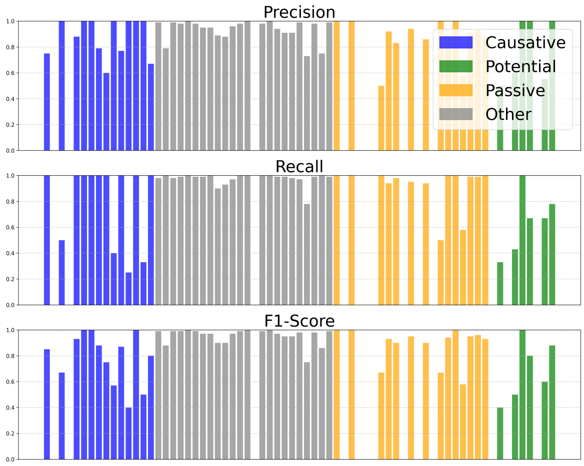


Figure 1: Baseline Model Metrics: Each bar represents a verb class, and the y-axis represents the score (precision, recall, F1), Generated with matplotlib (Hunter, 2007)

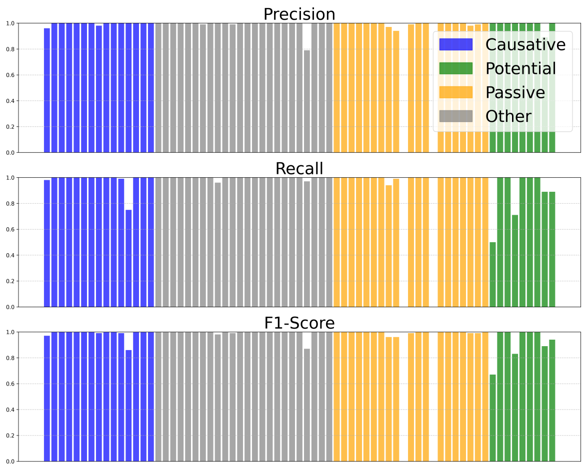


Figure 2: BiLSTM Model Metrics, Generated with matplotlib (Hunter, 2007)

5 Conclusion

The initial intuition was correct: the use of a model that is able to take in the full context of a sentence performs better than a model that only observes the verb and a few surrounding characters.

5.1 Problems and Improvements

As of now, the BiLSTM Model has not had its hyperparameters optimized (e.g. grid search), and the existing labeling script is not completely accurate, leading to the training data having some errors upon direct inspection, causing misclassifications by the model in terms of true correctness. For example, 「私は食べられた」(watashi wa taberareta) "I was able to eat" would be labeled as passive-past, despite context—or rather the lack of it—implying it is a potential-past verb.

Many verbs and sentences also had to be removed due to missing conjugation information from spaCy, and compound verbs (which are a combination of a nominal or participle verb form and another verb) are split where the second verb is ignored. 「です」(desu) verbs are also ignored, although classification of those is significantly easier for both a human and classification model due to being entirely rules based.

When classifying 「する」(suru) verbs that modify a noun, the noun itself is removed for simplicity and uniformity with how other verbs are conjugated as spaCy identifies 「する」(suru) as an auxillary verb, not a standalone verb due to the fact it often comes after a noun to modify it into a verb (e.g. 「勉強+する」(benkyou+suru) is the verb for "to study", where 「勉強」(benkyou) is the noun for "study")

In the future, I aim to fine-tune the hyperparameters of the BiLSTM Model (they were chosen arbitrarily), improve the existing classification script in Figure 2 , and add classification for 「です」 (desu) "to be" and compound verbs.

As mentioned previously, the method of distinguishing passive and potential verbs is not robust, as while in a majority of cases the presence of the particles 「が」(ga) or 「を」(wo) as markers of the noun directly before the verb and the presence of 「に」(ni) as an agent marker can indicate the type of verb, a manual inspection of the labeled dataset showed some exceptions. The semantic meaning of the noun being marked is also crucial (e.g. a noun indicating location/time is unlikely to be the agent of a passive verb, it is more likely to be the where/when of a potential verb), and the context and lack of it implies different classifications. Even if this comprehensive examination could be easily achieved, it still may not be fully accurate. The problem lies in the lack of a large, manually labeled dataset.

One potential alternative to rectify inconsistencies between potential and passive form classification is when labeling the dataset with one of these conjugations, instead of using particle information, translate the sentence to English first, then apply an existing English verb classifier model to label based off of that.

In order to add support for sentences containing compound verbs, retaining noun information in 「する」 verbs, etc., the tokenization process itself may have to change as well.

Finally, the last type of model to try for problem would be transformers, as their high performance in semantic analysis could be easily applicable to what is essentially a semantic classifier. If the previously mentioned translation approach was used to classify, a sentence transformer would be part of the pipeline for automated labeling as well.

References

José Fonseca. 2021a. Japanese verb conjugation scripts. Licensed under the MIT License.

Ricardo Cruz Fonseca. 2021b. Jamdict: A japanese-multilingual dictionary interface. https://jamdict.readthedocs.io/en/latest/. Accessed 2024-04-30.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, and 1 others. 2020. Array programming with numpy.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spacy: Industrial-strength natural language processing in python. https://spacy.io. Version 3.x.

J. D. Hunter. 2007. Matplotlib: A 2d graphics environment.

Taku Kudo. 2005. Mecab: Yet another part-of-speech and morphological analyzer. https://taku910.github.io/mecab/.

The pandas development team. 2020. pandas: powerful python data analysis toolkit. https://pandas.pydata.org. Version 1.x.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, and 1 others. 2019. Pytorch: An imperative style, high-performance deep learning library. https://pytorch.org. Version 1.x.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

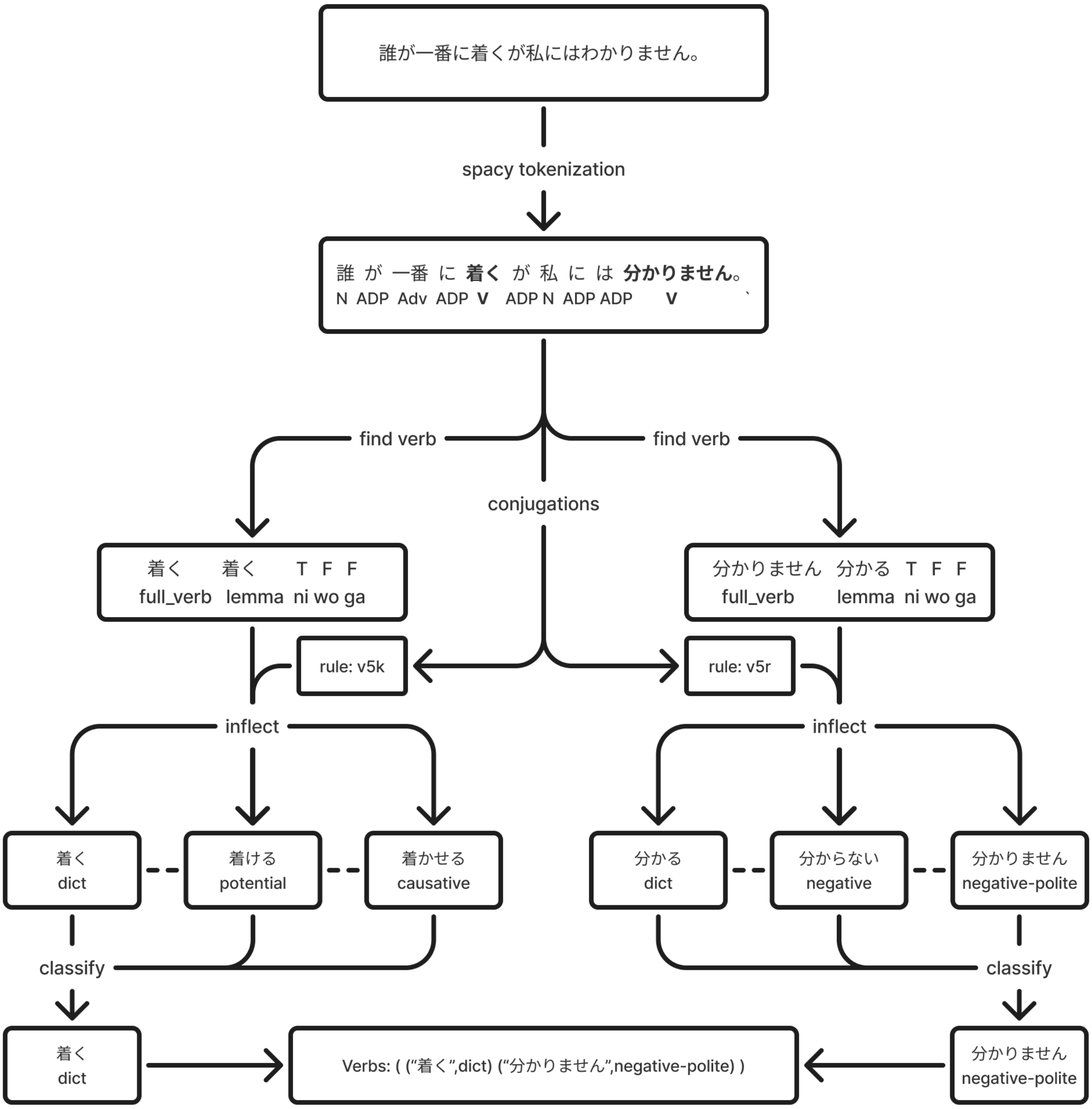
Tatoeba Project. 2021. Tanaka corpus. http://edrdg.org/wiki/index.php/Tanaka_Corpus.

Sketch Engine. n.d. Japanese mecab part-of-speech tagset. Accessed: 2025-05-01.

Guido van Rossum and the Python Development Team. 2009. Python 3 reference manual. Version 3.x.

Appendix

A Automated Classification Flowchart



B Japanese Verb Conjugation Examples

Conjugation Type	Example (Japanese)	Method
dict	食べる (taberu)	conjugation comparison
nominal	食べ (tabe)	conjugation comparison
past	食べた (tabeta)	conjugation comparison
past-conditional	食べたら (tabetara)	conjugation comparison
negative	食べない (tabenai)	conjugation comparison
negative-past	食べなかった (tabenakatta)	conjugation comparison
negative-past-conditional	食べなかったら (tabenakattara)	conjugation comparison
negative-nominal	食べなく (tabenaku)	conjugation comparison
negative-participle	食べないで (tabenaide)	conjugation comparison
passive	食べられる (taberareru)	particle check

Continued on next page

Table 3 – continued from previous page

Conjugation Type	Example (Japanese)	Method
passive-past	食べられた (taberareta)	particle check
passive-past-conditional	食べられたら (taberaretara)	particle check
passive-negative	食べられない (taberarenai)	particle check
passive-negative-past	食べられなかった (taberarenakatta)	particle check
passive-negative-past-conditional	食べられなかったら (taberarenakattara)	particle check
passive-negative-nominal	食べられなく (taberarenaku)	particle check
passive-negative-participle	食べられないで (taberarenaide)	particle check
passive-provisional-conditional	食べられれば (taberarereba)	particle check
passive-negative-provisional-conditional	食べられなければ (taberarenakereba)	particle check
passive-negative-colloquial	食べられなきゃ (taberarenakya)	particle check
passive-polite	食べられます (taberaremasu)	particle check
passive-past-polite	食べられました (taberaremashita)	particle check
passive-past-polite-conditional	食べられましたら (taberaremashitara)	particle check
passive-negative-polite	食べられません (taberaremasen)	particle check
passive-volitional-polite	食べられましょう (taberare mashou)	particle check
potential	食べられる (taberareru)	particle check
potential-past	食べられた (taberareta)	particle check
potential-past-conditional	食べられたら (taberaretara)	particle check
potential-negative	食べられない (taberarenai)	particle check
potential-negative-past	食べられなかった (taberarenakatta)	particle check
potential-negative-past-conditional	食べられなかったら (taberarenakattara)	particle check
potential-negative-nominal	食べられなく (taberarenaku)	particle check
potential-negative-participle	食べられないで (taberarenaide)	particle check
potential-provisional-conditional	食べられれば (taberarereba)	particle check
potential-negative-provisional-conditional	食べられなければ (taberarenakereba)	particle check
potential-negative-colloquial	食べられなきゃ (taberarenakya)	particle check
potential-polite	食べられます (taberaremasu)	particle check
potential-past-polite	食べられました (taberaremashita)	particle check
potential-past-polite-conditional	食べられましたら (taberaremashitara)	particle check
potential-negative-polite	食べられません (taberaremasen)	particle check
potential-volitional-polite	食べられましょう (taberare mashou)	particle check
potential-colloquial	食べれる (tabereru)	conjugation comparison
potential-colloquial-past	食べれた (tabereta)	conjugation comparison
potential-colloquial-past-conditional	食べれたら (taberetara)	conjugation comparison
potential-colloquial-negative	食べれない (taberenai)	conjugation comparison
potential-colloquial-negative-past	食べれなかった (taberenakatta)	conjugation comparison
potential-colloquial-negative-past-conditional	食べれなかったら (taberenakattara)	conjugation comparison
potential-colloquial-negative-nominal	食べれなく (taberenaku)	conjugation comparison
potential-colloquial-negative-participle	食べれないで (taberenaide)	conjugation comparison
potential-colloquial-provisional	食べれれば (taberereba)	conjugation comparison
potential-colloquial-negative-provisional	食べれなければ (taberenakereba)	conjugation comparison
potential-colloquial-negative-colloquial	食べれなきゃ (taberenakya)	conjugation comparison
potential-colloquial-polite	食べれます (taberemasu)	conjugation comparison
potential-colloquial-past-polite	食べれました (taberemashita)	conjugation comparison
potential-colloquial-past-polite-conditional	食べれましたら (taberemashitara)	conjugation comparison
potential-colloquial-negative-polite	食べれません (taberemasen)	conjugation comparison
potential-colloquial-volitional-polite	食べれましょう (taberemashou)	conjugation comparison
causative	食べさせる (tabesaseru)	conjugation comparison
causative-past	食べさせた (tabesaseta)	conjugation comparison
causative-past-conditional	食べさせたら (tabesasetara)	conjugation comparison
causative-negative	食べさせない (tabesasenai)	conjugation comparison
causative-negative-past	食べさせなかった (tabesasenakatta)	conjugation comparison
causative-negative-past-conditional	食べさせなかったら (tabesasenakattara)	conjugation comparison
causative-negative-nominal	食べさせなく (tabesasenaku)	conjugation comparison
causative-negative-participle	食べさせないで (tabesasenaide)	conjugation comparison
causative-provisional	食べさせば (tabesaseba)	conjugation comparison
causative-negative-provisional	食べさせなければ (tabesasenakereba)	conjugation comparison

Continued on next page

Table 3 – continued from previous page

Conjugation Type	Example (Japanese)	Method
causative-negative-colloquial	食べさせなきゃ (tabesasenakya)	conjugation comparison
causative-polite	食べさせます (tabesasemasu)	conjugation comparison
causative-past-polite	食べさせました (tabesasemashita)	conjugation comparison
causative-past-polite-conditional	食べさせましたら (tabesasemashitara)	conjugation comparison
causative-negative-polite	食べさせません (tabesasemasen)	conjugation comparison
causative-volitional-polite	食べさせましょう (tabesasemashou)	conjugation comparison
causative-passive	食べさせられる (tabesaserareru)	particle check
causative-passive-past	食べさせられた (tabesaserareta)	particle check
causative-passive-past-conditional	食べさせられたら (tabesaseraretara)	particle check
causative-passive-negative	食べさせられない (tabesaserarenai)	particle check
causative-passive-negative-past	食べさせられなかった (tabesaserarenakatta)	particle check
causative-passive-negative-past-conditional	食べさせられなかったら (tabesaserarenakattara)	particle check
causative-passive-negative-nominal	食べさせられなく (tabesaserarenaku)	particle check
causative-passive-negative-participle	食べさせられないで (tabesaserarenaide)	particle check
causative-passive-provisional	食べさせられれば (tabesaserarereba)	particle check
causative-passive-negative-provisional	食べさせられなければ (tabesaserarenakereba)	particle check
causative-passive-negative-colloquial	食べさせられなきゃ (tabesaserarenakya)	particle check
causative-passive-polite	食べさせられます (tabesaseraremasu)	particle check
causative-passive-past-polite	食べさせられました (tabesaseraremashita)	particle check
causative-passive-past-polite-conditional	食べさせられましたら (tabesaseraremashitara)	particle check
causative-passive-negative-polite	食べさせられません (tabesaseraremasen)	particle check
causative-passive-volitional-polite	食べさせられましょう (tabesaseraremashou)	particle check
causative-potential	食べさせられる (tabesaserareru)	particle check
causative-potential-past	食べさせられた (tabesaserareta)	particle check
causative-potential-past-conditional	食べさせられたら (tabesaseraretara)	particle check
causative-potential-negative	食べさせられない (tabesaserarenai)	particle check
causative-potential-negative-past	食べさせられなかった (tabesaserarenakatta)	particle check
causative-potential-negative-past-conditional	食べさせられなかったら (tabesaserarenakattara)	particle check
causative-potential-negative-nominal	食べさせられなく (tabesaserarenaku)	particle check
causative-potential-negative-participle	食べさせられないで (tabesaserarenaide)	particle check
causative-potential-provisional	食べさせられれば (tabesaserarereba)	particle check
causative-potential-negative-provisional	食べさせられなければ (tabesaserarenakereba)	particle check
causative-potential-negative-colloquial	食べさせられなきゃ (tabesaserarenakya)	particle check
causative-potential-polite	食べさせられます (tabesaseraremasu)	particle check
causative-potential-past-polite	食べさせられました (tabesaseraremashita)	particle check
causative-potential-past-polite-conditional	食べさせられましたら (tabesaseraremashitara)	particle check
causative-potential-negative-polite	食べさせられません (tabesaseraremasen)	particle check
causative-potential-volitional-polite	食べさせられましょう (tabesaseraremashou)	particle check
provisional-conditional	食べれば (tabereba)	conjugation comparison
negative-provisional-conditional	食べれなければ (taberenakereba)	conjugation comparison
negative-colloquial	食べなきゃ (tabenakya)	conjugation comparison
imperative	食べろ (tabero)	conjugation comparison
volitional	食べよう (tabeyou)	conjugation comparison
wish	食べたい (tabetai)	conjugation comparison
wish-past	食べたかった (tabetakatta)	conjugation comparison
wish-past-conditional	食べたかったら (tabetakattara)	conjugation comparison
wish-nominal	食べたく (tabetaku)	conjugation comparison
polite	食べます (tabemasu)	conjugation comparison
past-polite	食べました (tabemashita)	conjugation comparison
past-polite-conditional	食べましたら (tabemashitara)	conjugation comparison
negative-polite	食べません (tabemasen)	conjugation comparison
volitional-polite	食べましょう (tabemashou)	conjugation comparison