

SOFTWARE METAPAPER

ML-Ask: Open Source Affect Analysis Software for Textual Input in Japanese

Michal Ptaszynski¹, Pawel Dybala², Rafal Rzepka³, Kenji Araki³ and Fumito Masui⁴

¹ Software Development, Open Source Version Development, Department of Computer Science, Kitami Institute of Technology, Kitami, JP

² Software Development, Institute Of Middle and Far Eastern Studies, Faculty of International and Political Studies, Jagiellonian University, Kraków, PL

³ Software Development Supervision, Language Media Laboratory, Graduate School of Information Science and Technology, Hokkaido University, Sapporo, JP

⁴ Open Source Version Development Supervision, Department of Computer Science, Kitami Institute of Technology, Kitami, JP

Corresponding author: Michal Ptaszynski (ptaszynski@ieee.org)

We present ML-Ask – the first Open Source Affect Analysis system for textual input in Japanese. ML-Ask analyses the contents of an input (e.g., a sentence) and annotates it with information regarding the contained general emotive expressions, specific emotional words, valence-activation dimensions of overall expressed affect, and particular emotion types expressed with their respective expressions. ML-Ask also incorporates the Contextual Valence Shifters model for handling negation in sentences to deal with grammatically expressible shifts in the conveyed valence. The system, designed to work mainly under Linux and MacOS, can be used for research on, or applying the techniques of Affect Analysis within the framework Japanese language. It can also be used as an experimental baseline for specific research in Affect Analysis, and as a practical tool for written contents annotation.

Keywords: Affect Analysis; Sentiment Analysis; Open Source; Perl; Japanese

Funding statement: This research has been supported by: a Research Grant from the Nissan Science Foundation (years 2009–2010), The GCOE Program founded by Japan's Ministry of Education, Culture, Sports, Science and Technology (years 2009–2010), (JSPS) KAKENHI Grant-in-Aid for JSPS Fellows (Project Number: 22-00358) (years 2010–2012), (JSPS) KAKENHI Grant-in-Aid for Scientific Research (Project Number: 24600001) (years 2012–2015), (JSPS) KAKENHI Grant-in-Aid for Research Activity Start-up (Project Number: 25880003) (years 2013–2015), and (JSPS) KAKENHI Grant-in-Aid for Encouragement of Young Scientists (B) (Project Number: 15K16044) (years 2015–present, project estimated to end in March 2018).

(1) Overview

Introduction

Automatic analysis of user behavior and intentions has gained increasing interest through recent years. A large number of applications has been proposed from various sub-fields, including robotics, artificial intelligence (AI) or natural language processing (NLP). One of the most important tasks in such research and its applications is to properly recognize current state the user is in. Depending on application, the focus could be on different states of the user, such as user engagement in conversation (e.g., with a dialog agent [1]), user intention (e.g., to buy a certain product, or chose a specific migration route [2]), user attitude (e.g., toward a specific object, or the agent itself [3]), or user emotions (e.g., to choose different conversation strategy if the user is sad

or happy, etc. [4]). In many of those tasks techniques for Affect Analysis have proved to be effective. Affect Analysis refers to recognizing user affective states (emotions, moods, attitudes, etc.).

Several affect analysis systems have been proposed till now [7, 19, 9, 10, 14, 16, 41, 21, 28]. However, none of them has yet been released as an Open Source software. This paper presents the first Open Source system for text-based affect analysis of input in Japanese – ML-Ask. The system has been developed for several years and has matured enough to be released to the public. The system has already proved to be useful in multiple tasks and can be used for Affect Analysis in various research, as well as an experimental baseline for specific research in affect analysis and as a practical tool for annotation of written contents (such as user-generated contents on the Internet).

Background

Affect Analysis: Problem Definition

Text based Affect Analysis (AA) has been defined as a field focused on developing natural language processing (NLP) techniques for estimating the emotive aspect of text [5]. For example, Elliott [6] proposed a keyword-based Affect Analysis system applying an affect lexicon (including words like “happy”, or “sad”) with modifiers (words such as “extremely”, or “somewhat”). Liu et al. [7] presented a model of text-based affect sensing based on OMCS (Open-Mind Common Sense), a generic common sense database, with an application to e-mail interpretation. Alm et al. [8] proposed a machine learning method for Affect Analysis of fairy tales. Aman and Szpakowicz also applied machine learning techniques to analyze emotions expressed on blogs [19].

There have also been several attempts to achieve this goal for the Japanese language. For example, Tsuchiya et al. [9] tried to estimate emotive aspect of utterances with a use of an association mechanism. On the other hand, Tokuhisa et al. [10] as well as Shi et al. [11] used a large number of examples gathered from the Web to estimate user emotions.

Unfortunately, until now there have been no Open Source Affect Analysis systems. Although there exist several online demos, such as “Sentiment Analysis with Python NLTK Text Classification”,¹ or “Sentiment Analysis and Text Analytics Demo” by Lexalytics,² these refer to Sentiment Analysis, not Affect Analysis. In Sentiment Analysis the focus is usually put on determining emotion valence, or whether input (sentence, paragraph, product review, etc.) is of positive or negative valence. Affect Analysis is a task of much broader scope, focusing not only on the polarity of the input, but on particular emotion classes that are expressed by the input (joy, anger, fear, etc.).

To develop our software for Affect Analysis, we first needed to understand the phenomenon of how emotions are expressed in language. This phenomenon can be explained with the notion of the *emotive function of language*.

The Emotive Function of Language

Linguistic means used in conversation to inform interlocutors of emotional states are described by the *emotive function of language* (Jakobson, 1960) [39]. Ptaszynski (2006) [46] distinguished two kinds of its realizations in Japanese. The first one are emotive elements (or

emotemes) which indicate that emotions have been conveyed, but not detailing their specificity. This group is linguistically realized by interjections, exclamations, mimetic expressions, or vulgar language. The second are **emotive expressions** – parts of speech like nouns, verbs, adjectives, phrases or metaphors describing affective states. Nakamura (1993) [13] classified emotive expressions in Japanese into 10 emotion types said to be the most appropriate for the Japanese language. They can be translated as: joy, anger, gloom/sadness, fear, shame/shyness, fondness, dislike, excitement, relief and surprise (see **Table 2**).

Examples of sentences containing emotemes and/or emotive expressions are shown in **Table 1**. Examples (1) and (2) represent emotive sentences. (1) is an exclamative sentence, which is determined by the use of exclamative constructions *nante* (how/such a) and *nanda!* (exclamative sentence ending), and contains an emotive expression *kimochi ii* (feeling good/pleasant). (2) is also an exclamative. It is easily recognizable by the use of an interjection *iyaa*, an adjective in the function of interjection *sugoi* (great), and by the emphatic particle *-ne*. However, it does not contain any emotive expressions and therefore it is ambiguous whether the emotions conveyed by the speaker are positive or negative (or, in other words – such a sentence can be used in both positive and negative context). The examples (3) and (4) show non emotive sentences. Example (3), although containing a verb describing an emotional state *aishiteiru* (to love), is a generic statement and, if not put in a specific context, does not convey any emotions. Finally, (4) is a simple declarative sentence without any emotive value.

Definitions of Emotive Linguistic Features

Emotemes

Into the group of emotemes, structurally visualizable as textual representations of speech, Ptaszynski (2006) includes the following lexical and syntactical structures.

Exclamative utterance. Beijer (2002) [38] defines exclamative/emotive utterance, as every utterance in which the speaker is emotionally involved, and this involvement is expressed linguistically. The research on exclamatives in Japanese (Ono, 2002 [43]; Sasai, 2006 [47]) provides a wide scope of structures useful as features for our system. Some of the exclamative structures are: *nan (te/to/ka)-*, *-darou*, or *-da (yo/ne)*, partially

Table 1: Examples of sentences containing emotemes (underlined) and/or emotive expressions (bold type font). English translations were prepared to reflect both types if possible.

Example of a sentence (English translation)	emotemes	emotive expressions
(1) <i>Kyo wa <u>nante</u> kimochi ii hi <u>nanda!</u></i> (Today is <u>such a</u> nice day!)	yes	yes
(2) <i><u>Iyaa</u>, sore wa <u>sugoi</u> desu <u>ne!</u></i> (Whoa, what do you know!)	yes	no
(3) <i>Ryoushin wa minna jibun no kodomo wo aishiteiru.</i> (All parents love their children.)	no	yes
(4) <i>Kore wa hon desu.</i> (This is a book.)	no	no

Table 2: Examples from affect lexicon used in ML-Ask (N = noun, V = verb, Phr = phrase, Id = idiom, Adj = adjective, Adv = adverb).

喜 Joy	かたじけない <i>katajikenai</i> [Adj] grateful; 喜ばしい <i>yorokobashii</i> [Adj] delightful; 相好を崩す <i>soukou wo kuzusu</i> [Phr] be all smiles; 歓天喜地 <i>kantenkichi</i> [Id] extreme pleasure or satisfaction; わくわく <i>waku-waku</i> [Adv] be excited (with); 満足感 <i>manzoku-kan</i> [N] a feeling of satisfaction;
好 Fondness	恋い焦がれる <i>koikogareru</i> [V] be deeply in love, yearn; 気に入る <i>ki ni iru</i> [V] like something/somebody; 可愛がる <i>kawaigaru</i> [V] love, pet; 片思い <i>kata-omoi</i> [N] unrequited love; 大好き <i>dai-suki</i> [Adj] be very fond of; 愛欲 <i>aiyoku</i> [N] sexual desire, lust;
安 Relief	肩の荷が下りる <i>kata no ni ga oriru</i> [Phr] to have a weight removed from one's mind; 落ち着く <i>ochitsuku</i> [V] to calm down; らくらく <i>rakuraku</i> [Adv] very easily, effortlessly; ほっとする <i>hotto suru</i> [V] feel relieved; 一安心 <i>hitoanshin</i> [V] have peace of mind for a while
哀 Gloom	顔を曇らせる <i>kao wo kumoraseru</i> [Phr] cloud one's face; じいんと来る <i>jiin-to kuru</i> [Phr] be very touching; 一人ぼっち <i>hitoribocchi</i> [N] loneliness; 泣き叫ぶ <i>nakisakebu</i> [V] howl; 痛ましい <i>itamashii</i> [Adj] sad, pitiful; 思いやり <i>omoiyari</i> [N] sympathy, empathy;
厭 Dislike	むしゃくしゃ <i>musha-kusha</i> [Adv] irritated; 気に食わない <i>ki ni kuwanai</i> [Phr] be dissatisfied with; 虫酸が走る <i>mushizu ga hashiru</i> [Phr] be disgusted; 小憎らしい <i>konikurashii</i> [Adj] irritating; 陰気くさい <i>inki-kusai</i> [Adj] gloomy; やるせない <i>yarusenai</i> [Adj] downhearted;
怒 Anger	風向きが悪い <i>kazamuki ga warui</i> [Phr] be in bad mood; 食って掛かる <i>kuttekakaru</i> [V] fly at somebody; 怒り出す <i>okoridasu</i> [V] get into rage; ぶんぶん <i>punpun</i> [Adv] angrily, in huff; 息巻く <i>ikimaku</i> [V] storm at somebody; 不愉快 <i>fuyukai</i> [Adj] unpleasant;
怖 Fear	肝を冷やす <i>kimo wo hiyasu</i> [Phr] be frightened; 恐ろしい <i>osoroshii</i> [Adj] frightening, terrifying; おびえる <i>obieru</i> [V] be scared; 鳥肌 <i>torihada</i> [N] gooseflesh; 恐怖 <i>kyoufu</i> [N] fear; 不安 <i>fuan</i> [N] anxiety; 心配 <i>shinpai</i> [N] concern, anxiety;
恥 Shame	恥ずかしがる <i>hazukashigaru</i> [V] feel shy; 恥ずかしい <i>hazukashii</i> [Adj] disgraceful, shameful; はにかみ <i>hanikami</i> [N] bashfulness; 恥をかく <i>haji wo kaku</i> [V] be embarrassed; 赤面 <i>sekimen</i> [N] blush of shame; 恥 <i>haji</i> [N] shame; 俯く <i>utsumuku</i> [V] cast one's eyes down of shame;
昂 Excitement	飛び上がる <i>tobiagaru</i> [V] spring to one's feet; 気が急ぐ <i>ki ga seku</i> [Phr] to feel impatient; 隔靴搔痒 <i>kakkasouyou</i> [Phr] scratching through the sole of one's shoe; 感情的 <i>kanjouteki</i> [Adj] emotional, in heightened emotion; 興奮 <i>koufun</i> [N] excitement;
驚 Surprise	きょとんとする <i>kyotontosuru</i> [V] with a look of amazement; ショッキング <i>shokkingu</i> [Adj] shocking; 思いかけない <i>omoikakenai</i> [Adj] unexpected; びっくり仰天する <i>bikkurigyouten suru</i> [V] be absolutely astonished; びっくり <i>bikkuri</i> [N] surprise;

corresponding to *wh*-exclamatives in English (see the first sentence in **Table 1**).

Interjections immediately inform listeners/readers that the speaker/writer is in an emotional state, thus are typical emotemes. Interjections in Japanese include, e.g. *waa*, *yare-yare* or *iyaa*. See sentence (2) in **Table 1**.

Casual Speech. Casual speech is not an emoteme *per se*, however, many structures of casual speech are used when expressing emotions. Examples of casual language use could be modifications of adjective and verb endings *-ai* to *-ee*, like in the example: *Ha ga itee!* (My tooth hurts!), or abbreviations of forms *-noda* into *-nda*, like in the example: *Nani yattenda yo!?* (What the hell are you doing!?).

Gitaigo. Baba (2003) [37] distinguishes *gitaigo* (mimetic expressions) as emotemes specific for the Japanese language. Not all mimetics are emotive, but rather they can be classified into emotive mimetics (describing one's emotions), and sensation/state mimetics (describing manner and appearance). Examples of emotive *gitaigo* are: *iraira* (be/feel irritated), or *hiyahiya* (be in fear, nervous), like in the sentence: *Juugeki demo sareru n janai ka to omotte, hiyahiya shita ze.* (I thought he was gonna shoot me – I was petrified).

Emotive markers. This group contains punctuation marks used as a textual representations of emotive intonation features. The most obvious example is exclamation

mark “!”. In Japanese, marks like ellipsis “...”, prolongation marks, like “—”, or “~”, are also used to inform interlocutors that emotions have been conveyed (see **Table 1**).

Hypocoristics (endearments) in Japanese express emotions and attitudes towards an object by the use of diminutive forms of a name or status of the object (*Ai* [girl's name] vs *Ai-chan* [/endearment/]; *o-nee-san* [older sister] vs *o-nee-chan* [sis]). Example: *Saikin Oo-chan to Mit-chan ga bokura to karamu youni natte kita!!* (Oo-chan and Mit-chan have been palling around with us lately!!).

Vulgarity. The use of vulgarities usually accompanies expressing emotions. However, despite a general belief that vulgarities express only negative meaning, Ptaszynski (2006) noticed that they can be also used as expressions of strong positive feelings, and Sjöbergh (2006) [48] showed, that they can also be funny, when used in jokes, like in the example: *Mono wa mono dakedo, fuete komarimasu mono wa nanda—? Bakamono.* (A thing (*mono*) is a thing, but what kind of thing is bothersome if they increase? Idiots (*bakamono*)).

Emoticons. Emoticons have been used in online communication as generally perceived “emotion icons” (icons, or annotation markers, which inform readers of the writer's emotional state) for many years. Their numbers have developed depending on the language of use, letter input system, the kind of community they are used

in, etc. Popular emoticons include such examples, as “:-)” (smiling face), or “:-D” (laughing face). These are however not used by Japanese users. Emoticons, which are popular in Japanese communities, in contrast to the Western ones are usually unrotated and present faces, gestures, or postures from a point of view easily comprehensible to the reader. Some examples are: “ (^o^) ” (laughing face), “ (^_ ^) ” (smiling face), and “ (T_o_T) ” (crying face). They arose in Japan, where they were called *kaomoji*, in the 1980s and since then have been developed in a number of online communities.

Emotive Expressions

This group refers to a lexicon of expressions describing emotional states. Some examples include:

adjectives: *ureshii* (happy), *sabishii* (sad);
nouns: *aijō* (love), *kyōfu* (fear);
verbs: *yorokobu* (to feel happy), *aisuru* (to love);
fixed phrases/idioms: *mushizu ga hashiru* (give one the creeps [of hate]), *kokoro ga odoru* (one's heart is dancing [of joy]);
proverbs: *dohatsuten wo tsuku* (be in a towering rage), *ashi wo fumu tokoro wo shirazu* (be with one's heart up the sky [of happiness]);
metaphors/similes: *itai hodo kanashii* (sadness like a physical pain), *aijou wa eien no honoo da* (love is an eternal flame);

Such a lexicon can be used to express emotions, like in the first example in the **Table 1**, however, it can also be used to formulate non-emotive declarative sentences (third example in **Table 1**).

ML-Ask: Overview of the Software

Based on the linguistic approach towards emotions and the above-mentioned definition we constructed **ML-Ask** (*eMotive eLement and Expression Ana-lysis system*) software for automatic analysis and annotation of emotive information on written digital contents.

The emoteme databases for the system were gathered manually from linguistic literature and grouped into five types (code, reference research and number of gathered items in square, round and curly brackets, respectively):

1. **[EX]** Interjections and structures of exclamative and emotive-casual utterances (Nakamura, 1993 [42]; Oshima-Takane et al., 1995–1998 [44]; Tsuchiya, 1999 [50]; Ono, 2002 [43]). {462}
2. **[GI]** *Gitaigo* (Nakamura, 1993 [42]; Oshima-Takane et al., 1995–1998 [44]; Baba, 2003 [37]). {212}
3. **[HY]** Hypocoristics (Kamei et al., 1996 [40]). {4}
4. **[VU]** Vulgarities (Sjöbergh, 2008 [49]). {200}
5. **[EM]** Emotive marks (Kamei et al., 1996 [40]). {8}
6. **[EMOT]** Emoticons. For the detection and extraction of emoticons we applied in ML-Ask part of an algorithm of CAO, a *system for emotiCon Analysis and decOding of affective information*, developed by Ptaszynski et al. [27], which applies a refined set of 149 symbols statistically most frequently appearing in emoticons.

These databases were used as a core for ML-Ask.

Next, Nakamura's (1993) [42] dictionary was applied in the form of a database of emotive expressions (code: **[EMO-X]**, 2100 items in total). The breakdown with number of items per emotion type was as follows: joy {224}, anger {199}, gloom {232}, fear {147}, shame {65}, fondness {197}, dislike {532}, excitement {269}, relief {106}, surprise {129}. Some examples for each emotion type are shown in **Table 2**.

A textual input utterance/sentence is thus matched to the emoteme databases and emotive information is annotated. The software first determines whether an utterance is emotive (appearance of at least one emotive feature), extracts all emotive features from the sentence and describes the structure of the emotive utterance. The number of emotemes also expresses an emotive value, or the intensity of emotional load of the input. This is the software's main procedure for emotive information annotation of text collections. Next, in all utterances determined as emotive, the system searches for emotive expressions from the databases. The conceptual flow of the software procedures is represented on **Figure 1**.

Contextual Valence Shifters

To improve the system performance we also implemented Contextual Valence Shifters (CVS). The idea of CVS was first proposed by Polanyi and Zaenen [17, 45].

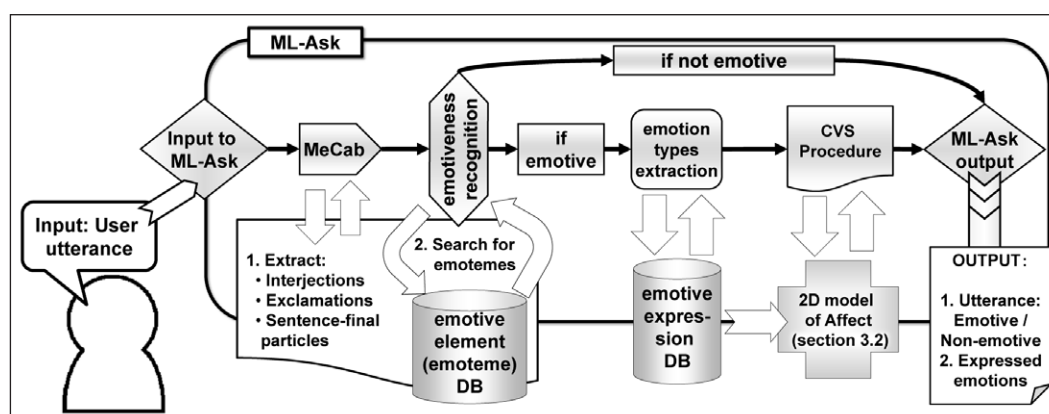


Figure 1: Conceptual flow of the ML-Ask software procedures.

They distinguished two kinds of CVS: negations and intensifiers. The group of negations contains words and phrases like “not”, “never”, and “not quite”, which change the valence polarity of the semantic orientation of an evaluative word they are attached to. The group of intensifiers contains words like “very”, “very much”, and “deeply”, which intensify the semantic orientation of an evaluative word. ML-Ask fully incorporates the negation type of CVS with a 108 syntactic negation structures. Examples of CVS negations in Japanese are structures such as: *amari -nai* (not quite-), *-to wa ienai* (cannot say it is-), or *-te wa ikenai* (cannot [verb]-). As for intensifiers, although ML-Ask does not include them as a separate database, most Japanese intensifiers are included in the emoteme database. The system also calculates emotive value, or emotional intensity of a sentence, on the basis of the number of emotemes in the sentence, thus the intensification is expressed with the emotive value. Two examples of valence shifting using Contextual Valence Shifters were represented in **Figure 2**.

Russell's 2-dimensional Model of Affect

Finally, the last distinguishable feature of ML-Ask is implementation of Russell's two dimensional affect space [18]. It assumes that all emotions can be represented in two dimensions: the emotion's valence or polarity (positive/negative) and activation (activated/deactivated).

An example of negative-activated emotion could be “anger”; a positive-deactivated emotion is, e.g., “relief”. The mapping of Nakamura's emotion types on Russell's two dimensions proposed by Ptaszynski et al. [21] was proved reliable in several research [21, 22, 27]. The mapping is represented in **Figure 3**. An example of ML-Ask output is represented in **Figure 4**.

Applications

ML-Ask has been applied to different tasks. Most commonly, the system was used to analyze user input in human-agent interaction [1, 3, 4, 15, 23, 24, 25, 26]. In particular, the analysis of user input was utilized in decision making support about which conversation strategy to choose (normal conversation or joke) [1], and in an automatic evaluation method for dialog agents [3]. ML-Ask was also used to help determine features specific to harmful entries in a task of cyberbullying detection [31]. ML-Ask supported with CAO was also applied in annotation of a large scale corpus (YACIS – Ameba blog corpus containing 5.6 bil. words), and together with a supporting Web-mining procedure in creation of a robust emotion object database [30]. In a recent research, ML-Ask has been used to detect emotions in mobile environment to help develop an accurate and user-adaptive emoticon recommendation system [32]. It has also been applied as a supporting procedure in automated ethical reasoning system [36].

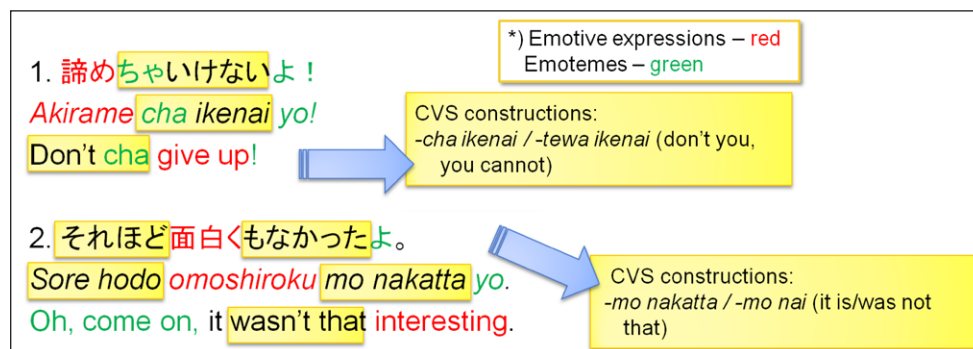


Figure 2: Examples of valence shifting using Contextual Valence Shifters.

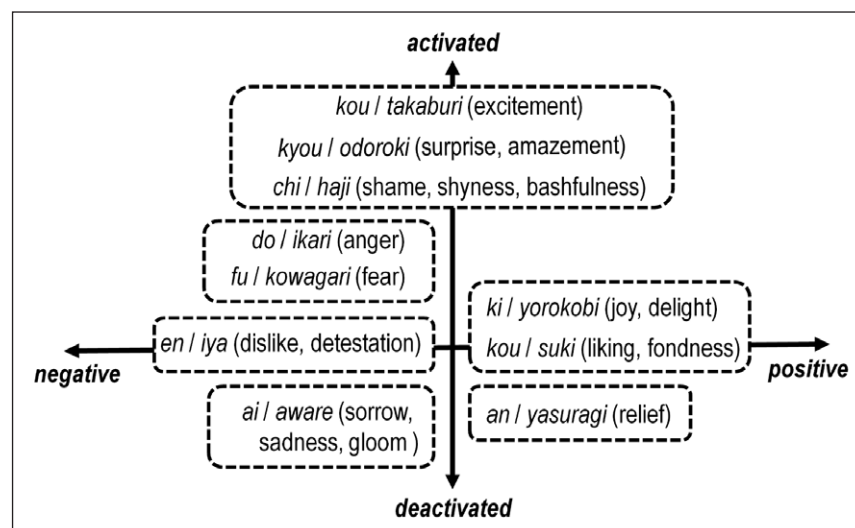


Figure 3: Mapping of Nakamura's classification of emotions on Russell's 2D space.

Implementation and architecture

ML-Ask was written in Perl Programming language.³ It works under Linux and macOS environments. Basic functionalities of ML-Ask can be launched in Windows environment as well, however, due to the differences in how some dependencies (especially MeCab: Yet Another Part-of-Speech and Morphological Analyzer)⁴ work under Windows, it is recommended to use ML-Ask under Linux or macOS operating systems.

Installation. System can be prepared for work in the following simple steps:

1. Download system files (either separately, or as the whole zipped package).
2. Unzip the package in case of downloading .zip file.
3. Install the dependencies:
 - Perl Programming Language: <https://www.perl.org/>.
 - MeCab: <http://taku910.github.io/mecab/>.
 - MeCab perl binding: <http://taku910.github.io/mecab/bindings.html>.
 - RE2 regex engine: <http://search.cpan.org/dist/re-engine-RE2/>.
4. Change directory to the folder containing all software files.

For the details of installation of MeCab and MeCab Perl binding refer to the above mentioned Web pages. Installation of RE2 regex engine can be easily done with the use of the CPAN perl module, or **cpan shell** – CPAN exploration and modules installation software (<http://www.cpan.org/>).

In case of problems with installation of RE2 engine, it is possible to run ML-Ask by deleting or commenting out from the main system file (`mlask[version_number].pl`) the line responsible for calling our the engine, namely:

```
use re::engine::RE2 -max_mem => 8<<23; #64MiB
```

To comment out a line in Perl one will put a hash symbol (#) at the beginning of the line, like below:

```
# use re::engine::RE2 -max_mem => 8<<23; #64MiB
```

Commenting out the RE2 engine will not influence the results, only the processing speed.

The system was designed to work in commandline with no additional GUI. This decision was made to reduce ML-Ask processing time to minimum to make the software capable of faster and more memory-efficient way of processing large files, and thus be applicable in BigData research.

The software can be launched in three modes: (1) Demo mode; (2) File processing mode; and (3) File processing mode with output to separate file. Below we describe those three modes.

Demo mode. In demo mode user launches the software using commandline and observes output appearing immediately on the screen (terminal). This mode is launched by typing `perl mlask[version_number].pl` in the commandline and pressing the [Enter] key, like below:

```
$ perl mlask4.3.pl
```

This command initializes the software and the user can input contents of their choice for further processing. An actual example of this input method and the following output is represented in **Figure 5**. Explanations of output interface is represented in detail in **Figure 6**. The sentence in the example is pronounced like:

Sentence:	なぜかレディーが*を見ると恐怖感じる(;´艸`)
Transliteration:	<i>Nazeka Lady Gaga wo miru to kyoufu kanjiru (;´艸`)</i>
Grammar:	Somehow Lady Gaga OBJ see COND fear feel EMOTICON
Translation:	Somehow Lady Gaga frightens me (;´艸`)

ML-Ask output: なぜかレディーが*を見ると恐怖感じる(;´艸`) sentence: emotive emotemes: EMOTICON: (;´艸`) emotions: (1), FEAR: 恐怖 2D: NEGATIVE, ACTIVE

Figure 4: Output example for ML-Ask.

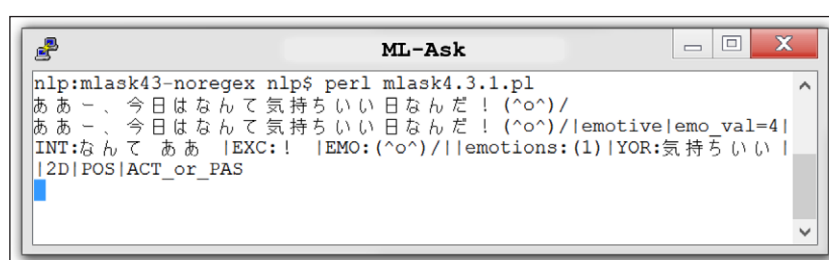


Figure 5: ML-Ask output.

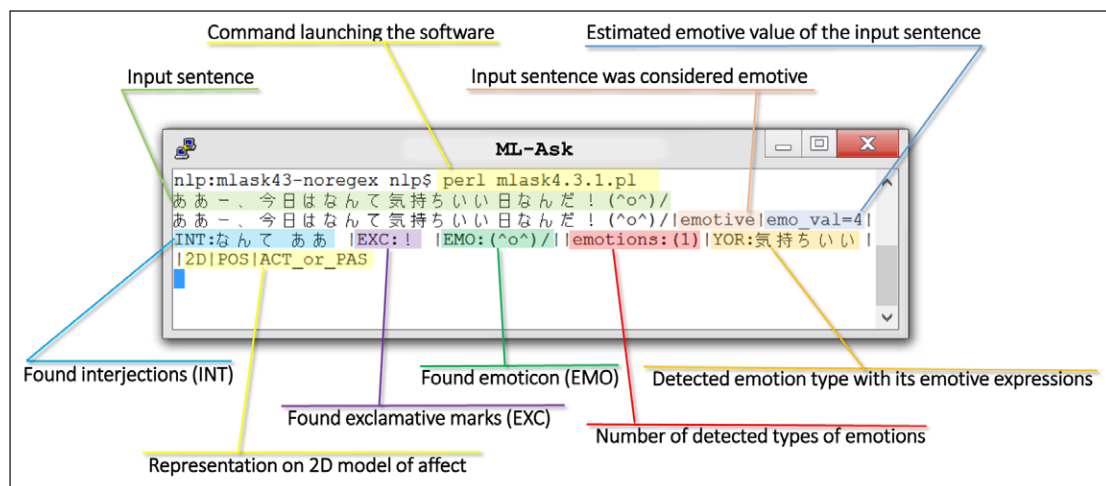


Figure 6: Explanations of ML-Ask output.

Aaa-, kyō wa nante kimochi ii hi nanda ! (^o^) /
which translates into:

“Aah..., what a pleasant weather it is today! (^o^) /”

The input sentence is considered emotive, and the estimated emotive value (`emo_val`) is calculated as 4. The found emotemes include interjections (INT), such as *nante* (“what a”), and *aa* (“Aah”), exclamation mark (EXC) “!”, and an emoticon (EMO) (^o^) /. One type of emotions is found in the sentence, namely joy, or *yorokobi* (YOR), with its representative emotive expression *kimochi ii* (“pleasant”). This emotion type is marked on Russell’s 2D affect space as being positive (POS) and either active or passive (ACT_or_PAS).

File rocessing mode w. STDOUT. In this mode the user, except typing the initial command, specifies the file they choose to process in the commandline and press the [Enter] key, like in the following:

```
$ perl mlassk[version_number].pl input_file.txt
```

The input file may contain multiple entries/sentences. ML-Ask processes and outputs each line separately. The output (all annotated sentences) appears on the screen (terminal). Since ML-Ask does not perform sentence segmentation, if one line contains multiple sentences, they will all be annotated as one document. This way the user has the choice of specifying themselves on what level they choose to analyze their data (document level, sentence level, chunk level, or phrase level).

File processing mode w. output to file. Finally, the user can send the output to an output file by typing in the commandline the following:

```
$ perl mlassk[version_number].pl input_file.txt > output_file.txt
```

This mode is useful and efficient for further processing and analysis, especially of very large files.

Software variant: ML-Ask-simple

ML-Ask was originally designed to analyze mostly conversation-like contents. In the first step of ML-Ask analysis the system specifies if a sentence is emotive or non-emotive. Analysis of particular emotion types is performed only on emotive sentences. A sentence is emotive if it contains at least one *emoteme*, or a marker of emotive

context. Emotemes are typical in conversations (in particular spontaneous conversations). Generally perceived narratives (blogs, fairytales, etc., often used in evaluation of affect analysis systems) contain at least two main types of sentences:

1. **descriptive sentences** for introduction of the main storyline, and
2. **dialogs** between characters of the narrative.

ML-Ask can be expected to deal with the second type of sentence. However, since emotemes rarely appear in descriptive sentences, the system would not precede to the recognition of particular emotion types for such sentences. Therefore, to allow ML-Ask deal with descriptive sentences as well we compiled a version of the system which excludes emotemes from the analysis and focuses primarily on analysis of emotion types. However, we retained the analysis of CVS and Russell’s emotion space. Since in this version of the system we simplified the analysis, we called it **ML-Ask-simple**. This variation of the software is launched in the same way as the original ML-Ask version.

Quality control

Quality control for this software has been done on two levels. One is the evaluation of the software as a concept, or a system – this is performed from a scientific viewpoint and the results are presented in scientific publications. This answers the question on, if the system works, how closely to fulfilling its goal does it work. Namely, how well does ML-Ask as a system detect and annotate affective states expressed in sentences. This part has been implemented from the start and is described in the first following subsection on “Evaluations.”

The second dimension of quality control refers to preparing ML-Ask as a working software. This includes regular revisions and improvements of the software code, performance improvements, when required also rewriting significantly large portions of code. Releasing the software as the Open Source and preparing the Open Source License also comes within this quality control spectrum,

as well as testing, benchmarking, and releasing every new update. These actions are described in the second subsection, “Release Related Actions.”

Evaluations

ML-Ask has been evaluated a number of times on different datasets and frameworks. In first evaluations, Ptaszynski et al. [12, 20, 21] focused on evaluating the system on separate sentences. For example, in [20], there were 90 sentences (45 emotive and 45 non-emotive) annotated by authors of the sentences (first-person standpoint annotations). On this dataset ML-Ask achieved 83% of balanced F-score for determining whether a sentence is emotive, 63% of human level of unanimity score for determining emotive value and 45% of balanced F-score for detecting particular emotion types. In [12] Ptaszynski et al. added annotations of third-party annotators and performed additional evaluation from the third-person standpoint. The evaluation showed that ML-Ask achieves better performance when supported by additional Web-mining procedure (not included in the OpenSource version) for extracting emotive associations from the Internet. This evaluation also showed that people are not ideal in determining emotions of other people. Additionally, in [21] Ptaszynski et al. performed an annotation of a Japanese BBS forum *2channel* with the use of ML-Ask. The dataset consisted of 1,840 sentences. The evaluation showed that there were two (out of ten) dominant emotion types (“dislike” and “excitement”) which were often expressed by sophisticated emoticons (multi-line ASCII-Art type), which the system could not detect. Without these two emotion types the system extracted other emotive tokens similarly to human annotators (90% of agreements).

After the above initial evaluations Ptaszynski et al. continued evaluation of ML-Ask on different datasets.

The system was most often evaluated on conversations, both between humans [1] and between human users and conversational agents [4, 3, 15, 23, 26]. In [1] Dybala et al. showed that ML-Ask presents comparable answers to human annotators when annotating conversations between people of different age and status (in particular young students vs. middle-aged businessmen). In other evaluations Ptaszynski et al. showed that the system performs comparably to humans when annotating human-agent dialogs. This was evaluated using only ML-Ask [3, 4], and ML-Ask confronted with the Web-mining procedure [23, 26]. Recently Ptaszynski et al. added also emoticon analysis system CAO to this evaluation [35].

Apart from the above evaluations, ML-Ask was also evaluated on blog contents. Firstly, in [25], using *Yahoo! blogs* (blogs.yahoo.co.jp) instead of the whole Web contents showed increased performance of the Web-mining procedure. Secondly, ML-Ask (alone and supported with emoticon analysis system CAO) was evaluated on YACIS, a corpus of blogs extracted from *Ameba blogs* (ameblo.jp). Finally, ML-Ask-simple was also recently evaluated using fairytales [33]. The evaluation showed performance of about 60.6% of accuracy, which shows that the system performs better on conversation-like contents, rather than on contents containing descriptive sentences. References to all evaluations of ML-Ask are represented in **Table 3**. Results of each evaluation were summarized in **Table 4**.

Release Related Actions

Quality assurance is also controlled for the software in the following ways.

Firstly, we perform regular code revisions and consider further improvements to the software code.

When required, we rewrite significant portions of code to improve performance and get rid of bugs.

Table 3: References describing evaluations and applications of ML-Ask.

Evaluations	Relevant research references
Separate Sentences	[12, 20, 21]
BBS	[21]
Conversations	[3, 23, 24, 25] [26, 35]
Blogs	[25, 29, 34, 35]
Fairytales	[33]
Applications	
Dialog agent:	
· Analysis of user input	[1, 3, 4, 15] [23, 24, 25, 26] [35]
· Decision making support	[1, 4]
· Automatic evaluation	[3]
Verification of emotion appropriateness	[23, 24, 25, 26] [35]
Corpus annotation	[34]
Emotion object database construction	[30]

Table 4: Results of each previous reported evaluation of ML-Ask. Evaluated functions are: Emo/N-emo = Emotive/non-emotive, EmoVal = Emotive value estimation, EmoType = Emotion type determination, Valence = Determining valence of emotion (positive/negative), Activ = Determining activation of emotion (active/passive), Engage = Estimation of emotional engagement in conversation. Evaluation metrics are: F1 = F-measure, U = Unanimity score, Acc = Accuracy, κ = Agreement with human (Kappa), ρ = Agreement with human (Pearson's "rho" ρ).

Paper No.	Year of publ.	Conference/Journal	Dataset used	Evaluated functions	Eval. metr.	Results reported
[20]	2008	Conference	Separate sentences	Emo/N-emo	F1	0.830
				EmoVal	U	0.630
				EmoType	F1	0.450
[12]	2009	Journal	Separate sentences	Emo/N-emo	F1	0.830
				EmoVal	U	0.630
				EmoType	F1	0.450
[23]	2009	Conference	Separate sent., Conversations	Emo/N-emo	F1	0.840
				EmoType	F1	0.450
[25]	2009	Conference	Conversations, Blogs	Valence	Acc	0.900
				EmoType	Acc	0.850
[21]	2009	Conference	Separate sentences	Emo/N-emo	Acc	0.900
				EmoType	F1	0.367
			BBS	Emo/N-emo	Acc	up to .75
				EmoType Annot.	κ	0.681
[3]	2010	Journal	Conversations	Engage	ρ	up to .597
[26]	2010	Journal	Separate sentences, Conversations	Emo/N-emo	F1	0.830
				EmoType	F1	0.470
				Valence	Acc	0.800
[29]	2012	Conference	Blogs	Emo/N-emo	Acc	0.988
				Valence/Activ	Acc	0.886
				EmoType	Acc	0.734
[34]	2013	Journal	Blogs	Emo/N-emo	F1	0.994
				Valence/Activ	F1	0.939
				EmoType	F1	0.847
[33]	2013	Journal	Fairytales	Valence/Activ	Acc	0.606
				EmoType	Acc	0.576
[35]	2013	Journal	Conversations, Blogs	Valence	Acc	0.600
				EmoType	Acc	0.680

Table 5: Example of benchmarking made for the current version of ML-Ask.

	Rate (iter/s)	mlask-4.2-simple (regex)	mlask-4.2 (regex)	mlask-4.3-simple (noregex)	mlask-4.3 (noregex)
dataset A (1 sentence from Figure 5)					
mlask-4.2-simple	206882875/s	–	–12%	–19%	–14%
mlask-4.2	234881024/s	14%	–	–8%	–2%
mlask-4.3	239674513/s	16%	2%	–6%	–
mlask-4.3-simple	254654171/s	23%	8%	–	6%
dataset B (90 sentences used in [20, 21])					
mlask-4.2-simple	126370709/s	–	–9%	–17%	–24%
mlask-4.2	139257523/s	10%	–	–9%	–17%
mlask-4.3-simple	152876500/s	21%	10%	–	–8%
mlask-4.3	166995436/s	32%	20%	9%	–

Every time prior to releasing a new updated version of the software we run manual tests on especially prepared data (e.g., sentences that should produce specific output) to check if the output is correct, and automatic benchmarking of software running time performed on both random data and especially prepared data. An example of such benchmarking made for the current software version is represented in **Table 5**.

The benchmark was performed on a MacPro with Intel Xeon E5 2.7 GHz, 12 cores (24 threads), DDR3 ECC 64 GB of memory, and 1 TB of PCI-e SSD.

The benchmarking tests were performed on two datasets. First contained only one sentence, namely the one represented in **Figure 5**. The second dataset contained 49 emotive sentences and 41 non-emotive sentences (also used, e.g., in [20], or [21]).

In summary, both versions of ML-Ask-4.3 were much faster than ML-Ask-4.2. However, whether ML-Ask-simple (both 4.2 and 4.3) was slower or faster than ML-Ask depends on several conditions. Firstly, each benchmark provides slightly different results and in other trials, for ML-Ask-4.3, both versions (simple and full) sometimes swapped places. The processed data also influences the generally perceived processing speed. For example, “-simple” versions perform Affect Analysis on all sentences without exception, while ML-Ask first performs pre-selection (emotive/non-emotive) of sentences for further processing. However, with the ability of processing one sentence in a fraction of a second it can be said that the software is well suited for large scale annotations of BigData corpora (as it was shown in, e.g., [34]).

A final touch for the quality control is providing a permanent contact email with the first developer (found in the “readme” file) to send any comments regarding the software and bug reports.

(2) Availability

Operating system

Linux (all modern distributions) and macOS (previously Mac OS X). Limited functionality on Windows (7 and higher).

Programming language

Perl 5.16 and higher.

Additional system requirements

No additional system requirements.

Dependencies

The following dependencies are required to run the software. The following dependencies may require other dependencies to install.

- MeCab: Yet Another Part-of-Speech and Morphological Analyzer
<http://taku910.github.io/mecab/#download>
- mecab-perl-binding
<http://taku910.github.io/mecab/#download>
- re::engine::RE2 Perl module (optional)

<http://search.cpan.org/~dgl/re-engine-RE2/lib/re/engine/RE2.pm>

The last dependency to install, namely **re::engine::RE2** Perl module is not necessary for running the software, although improves the processing speed. This is an improved regular expression engine for Perl. Although in the present version of ML-Ask we got rid of most of regular expressions in favor of other much faster pattern matching operations, there are still a few places which require the use of regular expression engine. The use of re2 will improve the performance for those places.

List of contributors

1. Michal Ptaszynski (Developer).
2. Pawel Dybala (Developer).
3. Rafal Rzepka (Software Development Supervisor).
4. Kenji Araki (Software Development Supervisor).
5. Fumito Masui (Open Source Version Development Supervisor).

Software location

Name: ML-Ask: Affect Analysis System

Software homepage: <http://arakilab.media.eng.hokudai.ac.jp/~ptaszynski/repository/mlask.htm>

Persistent identifier: <https://doi.org/10.5281/zenodo.556476>

Licence: New BSD License

Publisher: Michal Ptaszynski

Version published: 4.2, 4.3, 4.3.1

Date published: 27/09/2011

Code repository GitHub

Name: mlask

Persistent identifier: <https://github.com/ptaszynski/mlask>

Licence: New BSD License

Date published: 28/09/2016

Language

A “readme” file in English is included in the software package and contains a short introduction, overview of the software, installation and usage manual, copyrights and list of contributors, as well as references to scientific papers describing the system on which the software was built.

Text of the Open Source license (New BSD License) is provided together with the software in English.

Language the software is programmed to process is Japanese.

(3) Reuse potential

The typical usage of ML-Ask is to detect whether a written entry in Japanese (a sentence, a written utterance, a tweet, a blog post, etc.) was produced with an emotional state and specify what expressions of emotions exactly this entry contains. However, as mentioned previously in section “Applications,” ML-Ask has been applied to a multitude of tasks related to its main functionality.

Except generally analyzing user input in human-agent interaction, the information provided by ML-Ask was used to determine which conversation strategy to choose (normal conversation or humorous response).

The information on affective states expressed by the user were also used as information on how the user feels about the dialog system they interact with. As the support for decision making systems, ML-Ask has been used to detect emotions in sentences (email, messages, etc.) entered by the user in mobile application to help recommend emoticons that would fit the emotional atmosphere of the message.

Affect analysis of Internet entries with ML-Ask occurred effective in determining features specific to harmful entries in a task of cyberbullying detection. Since affective information provided by ML-Ask is very rich, this could indicate that such information, could be useful in determining distinguishable features for other tasks related to affect and sentiment analysis, or even generally perceived binary text classification.

Since the performance of ML-Ask is sufficient, as it was indicated by applying it to annotate a large scale collection of blogs, corpus annotation with affective information is also one of potential reuse purposes.

Notes

¹ <http://text-processing.com/demo/sentiment/>

² <https://www.lexalytics.com/demo>

³ <https://www.perl.org/>

⁴ <http://taku910.github.io/mecab/>

Competing Interests

The authors have no competing interests to declare.

References

1. **Dybala, P, Ptaszynski, M, Rzepka, R and Araki, K** 2009 "Activating Humans with Humor? A Dialogue System that Users Want to Interact With", *IEICE Transactions on Information and Systems*, E92-D(12): 2394–2401, (December). DOI: <https://doi.org/10.1587/transinf.E92.D.2394>
2. **Schrempf, O C and Hanebeck, U D** 2005 "A generic model for estimating user-intentions in human-robot cooperation." In: *Proceedings of the 2nd International Conference on Informatics in Control, Automation and Robotics, ICINCO*, Vol. 5. 2005.
3. **Ptaszynski, M, Dybala, P, Rzepka, R and Araki, K** 2010 "An Automatic Evaluation Method for Conversational Agents Based on Affect-as-Information Theory". *Journal of Japan Society for Fuzzy Theory and Intelligent Informatics*, 22(1): 73–89, (February). DOI: <https://doi.org/10.3156/jsoft.22.73>
4. **Dybala, P, Ptaszynski, M, Higuchi, S, Rzepka, R and Araki, K** 2008 Humor Prevails! – Implementing a Joke Generator into a Conversational System. *LNAI*, 5360: 214–225. DOI: https://doi.org/10.1007/978-3-540-89378-3_21
5. **Grefenstette, G, Qu, Y, Shanahan, J G and Evans, D A** 2004 "Coupling Niche Browsers and Affect Analysis for an Opinion Mining", In: *Proceedings of RIAO-04*, pp. 186–194.
6. **Elliott, C** 1992 "The Affective Reasoner – A Process Model of Emotions in a Multi-agent System", PhD thesis, Northwestern University, May 1992. The Institute for the Learning Sciences, Technical Report No. 32.
7. **Liu, H, Lieberman, H and Selker, T** 2003 "A Model of Textual Affect Sensing using Real-World Knowledge", In: *Proceedings of IUI 2003*, pp. 125–132. DOI: <https://doi.org/10.1145/604045.604067>
8. **Alm, C O, Roth, D and Sproat, R** 2005 "Emotions from text: machine learning for text based emotion prediction", In: *Proc. of HLT/EMNLP*, pp. 579–586. DOI: <https://doi.org/10.3115/1220575.1220648>
9. **Tsuchiya, S, Yoshimura, E, Watabe, H and Kawaoka, T** 2007 "The Method of the Emotion Judgment Based on an Association Mechanism", *Journal of Natural Language Processing*, 14(3): 219–238. DOI: https://doi.org/10.5715/jnlp.14.3_219
10. **Tokuhisa, R, Inui, K and Matsumoto, Y** 2008 "Emotion Classification Using Massive Examples Extracted from the Web", In: *Proc. of Coling 2008*, pp. 881–888, 2008. DOI: <https://doi.org/10.3115/1599081.1599192>
11. **Shi, W, Rzepka, R and Araki, K** 2008 "Emotive Information Discovery from User Textual Input Using Causal Associations from the Internet" [In Japanese], *FIT2008*, pp. 267–268.
12. **Ptaszynski, M, Dybala, P, Shi, W, Rzepka, R and Araki, K** 2009 "A System for Affect Analysis of Utterances in Japanese Supported with Web Mining", *Journal of Japan Society for Fuzzy Theory and Intelligent Informatics*, Special Issue on Kansei Retrieval, 21(2): 30–49 (194–213), (April).
13. **Nakamura, A** 1993 "Kanjo hyogen jiten" [Dictionary of Emotive Expressions] (in Japanese), Tokyodo Publishing, Tokyo.
14. **Wilson, T and Wiebe, J** 2005 Annotating Attributions and Private States. *Proceedings of the ACL Workshop on Frontiers in Corpus Annotation II*. pp. 53–60. DOI: <https://doi.org/10.3115/1608829.1608837>
15. **Dybala, P, Ptaszynski, M, Rzepka, R and Araki, K** Extracting Dajare Candidates from the Web – Japanese Puns Generating System as a Part of Humor Processing Research, In: *The Proceedings of the First International Workshop on Laughter in Interaction and Body Movement (LIBM'08)*, pp. 46–51, Asahikawa, Japan, June 2008.
16. **Wiebe, J, Wilson, T and Cardie, C** 2005 "Annotating expressions of opinions and emotions in language", *Language Resources and Evaluation*, 39(2–3): 165–210. DOI: <https://doi.org/10.1007/s10579-005-7880-9>
17. **Zaenen, A and Polanyi, L** 2006 "Contextual Valence Shifters", In: *Computing Attitude and Affect in Text*, Shanahan, J G, Qu, Y and Wiebe, J (eds.), Springer Verlag, Dordrecht, The Netherlands, pp. 1–10.
18. **Russell, J A** 1980 "A circumplex model of affect", *J. of Personality and Social Psychology*, 39(6): 1161–1178. DOI: <https://doi.org/10.1037/h0077714>
19. **Aman, S and Szpakowicz, S** 2007 "Identifying Expressions of Emotion in Text", In: *Proceedings of the 10th International Conference on Text, Speech, and*

- Dialogue (TSD-2007)*, Plzen, Czech Republic, Lecture Notes in Computer Science (LNCS), Springer-Verlag. DOI: https://doi.org/10.1007/978-3-540-74628-7_27
20. **Ptaszynski, M, Dybala, P, Shi, W, Rzepka, R and Araki, K** 2008 Disentangling emotions from the Web. Internet in the service of affect analysis. In: *Proceedings of KEAS'08*, pp. 51–56.
 21. **Ptaszynski, M, Dybala, P, Rzepka, R and Araki, K** 2009 Affecting Corpora: Experiments with Automatic Affect Annotation System – A Case Study of the 2channel Forum, *The Conference of the Pacific Association for Computational Linguistics (PACLING-09)*, September 1–4, 2009, Hokkaido University, Sapporo, Japan, pp. 223–228. DOI: <https://doi.org/10.13140/RG.2.2.13098.21444>
 22. **Ptaszynski, M, Dybala, P, Shi, W, Rzepka, R and Araki, K** 2009 “Towards Context Aware Emotional Intelligence in Machines: Computing Contextual Appropriateness of Affective States”, In: *Proceedings of Twenty-first International Joint Conference on Artificial Intelligence (IJCAI-09)*, Pasadena, California, USA, pp. 1469–1474.
 23. **Ptaszynski, M, Dybala, P, Shi, W, Rzepka, R and Araki, K** 2009 “Towards Context Aware Emotional Intelligence in Machines: Computing Contextual Appropriateness of Affective States”, In: *Proceedings of Twenty-first International Joint Conference on Artificial Intelligence (IJCAI-09)*, pp. 1469–1474.
 24. **Ptaszynski, M, Dybala, P, Shi, W, Rzepka, R and Araki, K** “Shifting Valence Helps Verify Contextual Appropriateness of Emotions”, *The IJCAI-09 Workshop on Automated Reasoning about Context and Ontology Evolution (ARCOE-09)*, in Working Notes of Twenty-first International Joint Conference on Artificial Intelligence (IJCAI-09), Pasadena, California, USA, 2009, pp. 19–21.
 25. **Ptaszynski, M, Dybala, P, Shi, W, Rzepka, R and Araki, K** “Conscience of Blogs: Verifying Contextual Appropriateness of Emotions Basing on Blog Contents”, *The Fourth International Conference on Computational Intelligence (CI 2009)*, August 17–19, 2009, Honolulu, Hawaii, USA, pp. 1–6.
 26. **Ptaszynski, M, Dybala, P, Shi, W, Rzepka, R and Araki, K** “Contextual Affect Analysis: A System for Verification of Emotion Appropriateness Supported with Contextual Valence Shifters”, *International Journal of Biometrics*, 2(2): 134–154, 2010. DOI: <https://doi.org/10.1504/IJBM.2010.031793>
 27. **Ptaszynski, M, Maciejewski, J, Dybala, P, Rzepka, R and Araki, K** 2010 “CAO: Fully Automatic Emoticon Analysis System”, In: *Proc. of the 24th AAAI Conference on Artificial Intelligence (AAAI-10)*, pp. 1026–1032.
 28. **Ptaszynski, M, Maciejewski, J, Dybala, P, Rzepka, R and Araki, K** 2010 “CAO: A Fully Automatic Emoticon Analysis System Based on Theory of Kinesics”, *IEEE Transactions on Affective Computing*, 1(1): 46–59. DOI: <https://doi.org/10.1109/T-AFFC.2010.3>
 29. **Ptaszynski, M, Dybala, P, Rzepka, R, Araki, K and Momouchi, Y** 2012 “Annotating Affective Information on 5.5 Billion Word Corpus of Japanese Blogs”, In: *Proceedings of The Eighteenth Annual Meeting of The Association for Natural Language Processing (NLP-2012)*, Hiroshima, Japan, March 13–16, pp. 405–408.
 30. **Ptaszynski, M, Rzepka, R, Araki, K and Momouchi, Y** 2012 “A Robust Ontology of Emotion Objects”, In: *Proceedings of The Eighteenth Annual Meeting of The Association for Natural Language Processing (NLP-2012)*, Hiroshima, Japan, March 13–16.
 31. **Ptaszynski, M, Dybala, P, Matsuba, T, Masui, F, Rzepka, R, Araki, K and Momouchi, Y** 2010 “In the Service of Online Order: Tackling Cyber-Bullying with Machine Learning and Affect Analysis”, *International Journal of Computational Linguistics Research*, 1(3): 135–154, 2010.
 32. **Urabe, Y, Rzepka, R and Araki, K** 2014 Emoticon Recommendation System to Richen Your Online Communication, *International Journal of Multimedia Data Engineering and Management*, 5(1): 14–33, January–March 2014.
 33. **Ptaszynski, M, Dokoshi, H, Oyama, S, Rzepka, R, Kurihara, M, Araki, K and Momouchi, Y** 2013 “Affect Analysis in Context of Characters of Narratives”, *Expert Systems With Applications*, 40(1): 168–176, January 2013. DOI: <https://doi.org/10.1016/j.eswa.2012.07.025>
 34. **Ptaszynski, M, Rzepka, R, Araki, K and Momouchi, Y** 2013 “Automatically Annotating A Five-Billion-Word Corpus of Japanese Blogs for Sentiment and Affect Analysis”, *Computer Speech and Language (CSL)*, Elsevier, 2013. (to appear)
 35. **Ptaszynski, M, Mazur, M, Dybala, P, Rzepka, R, Araki, K and Momouchi, Y** 2013 “Towards Computational Fronesis: Verifying Contextual Appropriateness of Emotions”, *International Journal of Distance Education Technologies (IJDET)*, Special Issue on Emotional Intelligence for Online Learning, 11(2), 2013. DOI: <https://doi.org/10.4018/jdet.2013040102>
 36. **Komuda, R, Ptaszynski, M, Momouchi, Y, Rzepka, R, and Araki, K** 2010 “Machine Moral Development: Moral Reasoning Agent Based on Wisdom of Web-Crowd and Emotions”, *Int. Journal of Computational Linguistics Research*, 1(3): 155–163.
 37. **Baba, J** 2003 Pragmatic function of Japanese mimetics in the spoken discourse of varying emotive intensity levels. *Journal of Pragmatics*, 35(12): 1861–1889, Elsevier. DOI: [https://doi.org/10.1016/S0378-2166\(03\)00048-1](https://doi.org/10.1016/S0378-2166(03)00048-1)
 38. **Beijer, F** 2002 The syntax and pragmatics of exclamations and other expressive/emotional utterances. *Working Papers in Linguistics 2*, The Dept. of English in Lund.
 39. **Jakobson, R** 1960 Closing Statement: Linguistics and Poetics. *Style in Language*, 350–377, The MIT Press.
 40. **Kamei, T, Kouno, R and Chino, E** (eds.) 1996 *The Sanseido Encyclopedia of Linguistics*, VI, Sanseido.
 41. **Minato, J, Bracewell, D B, Ren, F and Kuroiwa, S** 2006 Statistical Analysis of a Japanese Emotion Corpus for Natural Language Processing. *LNCS*, 4114.
 42. **Nakamura, A** 1993 *Kanjo hyogen jiten* [Dictionary of Emotive Expressions] (in Japanese), Tokyodo Publishing.

43. **Ono, H** 2002 *An emphatic particle DA and exclamatory sentences in Japanese*. Univ. of California, Irvine.
44. **Oshima-Takane, Y** and **MacWhinney, B** (eds.), 1995–1998 *CHILDES Manual for Japanese*, Shirai, H, Miyata, S and Naka, N (Rev.), McGill University, The JCHAT Project.
45. **Polanyi, L** and **Zaenen, A** 2004 Contextual Valence Shifters, *AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*.
46. **Ptaszynski, M** 2006 *Boisterous language. Analysis of structures and semiotic functions of emotive expressions in conversation on Japanese Internet bulletin board forum '2channel'*, M.A. Dissertation, UAM, Poznan.
47. **Sasai, K** 2006 The Structure of Modern Japanese Exclamatory Sentences: On the Structure of the Nanto-Type Sentence. *Studies in the Japanese language*, 2(1): 16–31.
48. **Sjöbergh, J** 2006 Vulgarities are fucking funny, or at least make things a little bit funnier. *Technical Report of KTH*, Stockholm.
49. **Sjöbergh, J** and **Araki, K** 2008 A Multi-Lingual Dictionary of Dirty Words, *LREC*.
50. **Tsuchiya, N** 1999 Taiwa ni okeru kandoshi, iiyodomi no togoteki seishitsu ni tsuite no kosatsu. [Statistical observations of interjections and faltering in discourse] (in Japanese), *SIG-SLUD-9903-11*. 1999.

How to cite this article: Ptaszynski, M, Dybala, P, Rzepka, R, Araki, K and Masui, F 2017 ML-Ask: Open Source Affect Analysis Software for Textual Input in Japanese. *Journal of Open Research Software*, 5: 16, DOI: <https://doi.org/10.5334/jors.149>

Published: 22 September 2016

Accepted: 10 May 2017

Published: 07 June 2017

Copyright: © 2017 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.