Normalizing Emotion-Driven Acronyms towards Decoding Spontaneous Short Text Messages

Bizhanova Aizhan and Atsushi Fujii
Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan

Keywords: Natural Language Processing, Word Sense Disambiguation, Text Normalization, Social Networking Service, Information Retrieval, Acronym, Emotion.

Abstract: Reflecting the rapid growth in the use of Social Networking Services (SNSs), it has of late become popular for users to share their feelings, impression, and opinions with each other, about what they saw or experienced, rapidly by means of short text messages (SMS). This trend has let a large number of users consciously or unconsciously use emotion-bearing words and also acronyms to reduce the number of characters to type. We have noticed this new emerging category of language unit, namely "Emotion-Driven Acronyms (EDAs)". Because by definition, each acronym consists of less characters than its original full form, the acronyms for different full forms often coincidently identical. Consequently, the misuse of EDAs substantially decreases the readability of messages. Our long-term research goal is to normalize text in a corrupt language into the canonical one. In this paper, as the first step towards the exploration of EDAs, we focus only on the normalization for EDAs and propose a method to disambiguate the occurrence of an EDA that corresponds to different full forms depending on the context, such as "smh (so much hate / shaking my head)". We also demonstrate what kind of features are effective in our task experimentally and discuss the nature of EDAs from different perspectives.

1 INTRODUCTION

With the rapid technological development facilitating easy access to the Internet via personal mobile devices, the number of registered users on Social Networking Services (SNSs) has been growing. Specifically, this trend has been remarkable for Twitter.

In Twitter, the length of each submission, or termed "tweet", is restricted up to 140 words and for that reason exchanging tweets with other users is like an informal chat rather than sending a letter enclosed in an envelope. As a result, the content of each message tends to be spontaneous and emotional, namely your instinctive impression or psychological response to what you received through any of the five senses.

Looking at a negative side of this trend, a representative example is associated with corruption of languages. Due to its nature, each tweet is rarely revised before submission and so often contain much noise, such as typographical and grammatical errors. Additionally, an unofficial short form of a certain expression is created, specifically for long and high frequent phrases, to convey as much information as possible by a limited number of words and also improve the efficiency of texting. Consequently, idiomatic or fixed phrases including emotion-bearing words ("so much hate", "oh my god") are likely to be replaced with an acronym, such as "smh" and "omg", respectively. We have noticed such an emerging trend and also believe it is worth exploring the properties specifically inherent in a group of these acronyms containing one or more emotion-bearing words, namely "Emotion-Driven Acronyms (EDAs)".

The definition for acronym can be different, depending on the source dictionary, and so we will not argue regarding the difference between acronym and its synonym, such as abbreviation and initialism.

Acronyms often are compound nouns, and function phrases, that keep the rhetorical structure of sentences well-formed, such as "ASAP (as soon as possible)", "FYI (for your information)", and "WRT (with respect to)". EDA also functions as an interjection, such as "omg (oh my god)".

Our research interests are associated with text normalization, which is general term for Natural Language Processing (NLP) intended to translate text

Aizhan, B. and Fujii, A.
Normalizing Emotion-Driven Acronyms towards Decoding Spontaneous Short Text Messages.
DOI: 10.5220/0007407707310738
In *Proceedings of the 11th International Conference on Agents and Artificial Intelligence (ICAART 2019)*, pages 731-738
ISBN: 978-989-758-350-6

in a corrupt language into the canonical one. Whereas our long-term goal is the text normalization for any corruption, as the first step of research, in this paper we focus only on disambiguation of EDAs. The contribution of our research is that this paper is the first awareness and exploration of EDAs in the NLP and its related communities, such as Information Retrieval (IR).

2 RELATED WORK

For NLP and its related research communities, such as IR, the frequent use of newly created acronyms leads to the vocabulary mismatch (VM) and out-of-vocabulary (OOV) problems. Due to VM, the occurrences of constituent words in EDAs would be underestimated, whereas the occurrences of EDAs are coincidently identical to an existing word, such as "kiss (keep it simple stupid)", the occurrences of that word would be overestimated.

Intended to alleviate the problems associated with acronyms, a large number of methods that can potentially contribute to normalization of acronyms have been proposed, which can roughly be classified into three groups. Whereas acronym identification is intended to detect the occurrence of an acronym in a document, acronym expansion is intended to recover the full form of an acronym in question. Additionally, acronym disambiguation is an application of word sense disambiguation (WSD), which is intended to select the most plausible meaning for each occurrence of a word associated with a lexical ambiguity, namely homonymy and polysemy. The ambiguity associated with acronyms is due to that the spelling of more than one expression coincidentally is the same, and thus is more similar to homonymy than polysemy.

We review several research references associated with the acronym disambiguation, from different perspectives.

As with a large number of acronyms in technical terms, certain acronyms are often domain-specific. (Bracewell, Russel and Wu, 2006) proposed methods for identification, expansion, and disambiguation of acronyms in biomedical texts. Naïve Bayesian classifier-based hybrid approach was used for the identification and expansion tasks, two expansion tasks, namely local and global expansions are performed. For the local expansion, windowing and longest common subsequence is used to generate candidates of expansion. For the global expansion, an external acronym database UMLS is used.

(Barua and Patel, 2016) disambiguate acronyms in the short text messages by producing an acronym dictionary, which is updated by consistently monitoring the media.

(Moon, McInnes and Melton, 2015) explored acronym disambiguation in the healthcare domain. With the adoption of electronic health record systems and consistent usage of electronic clinical documents, the use of acronyms substantially increased. One of the technological challenges was selecting effective features for the disambiguation for acronyms.

(Duque, Martinez-Romo and Araujo, 2016) proposed the use of co-occurrence graphs containing biomedical concepts and textual information for WSD targeting the biomedical domain.

There are several pieces of work that use semantic similarity between contexts.

(Li, Ji and Yan, 2015) used the word embedding to calculate the semantic similarity between words.

(Sridhar, 2015) proposed a method to learn phrase normalization lexicons by training distributed representations over compound words.

(Boguraev, Chu-Carroll, Ferrucci, Levas and Prager, 2015) used a set of words that co-occur with a target acronym within a specific proximity as a pseudo (surrogate) document and also produced an index so that the disambiguation can be recast as searching for the words frequently co-occur with query words in the same context.

To deal with ambiguous acronyms in scientific papers (Charbonnier and Wartena, 2018) proposed a method, which learns word embeddings for all words in the corpus and compare the averaged context vector of the words in the expansion of an acronym.

Before disambiguating acronyms, we need to expand them to identify their full forms. This task, abbreviation expansion, has been addressed. Whereas (Sproat, Black, Chen, Kumar, Ostendorf and Richards, 2001) proposed both supervised and unsupervised approaches, (Xue, Yin and Davison, 2011) combined orthographic, phonetic, contextual factors for expanding acronyms in a channel model.

3 DISAMBIGUATION METHOD

As we saw in Section 2, a) identification of the candidate full forms for an acronym in question, b) modelling of each candidate, and c) matching of a query to the correct candidate are major factors for disambiguation systems.

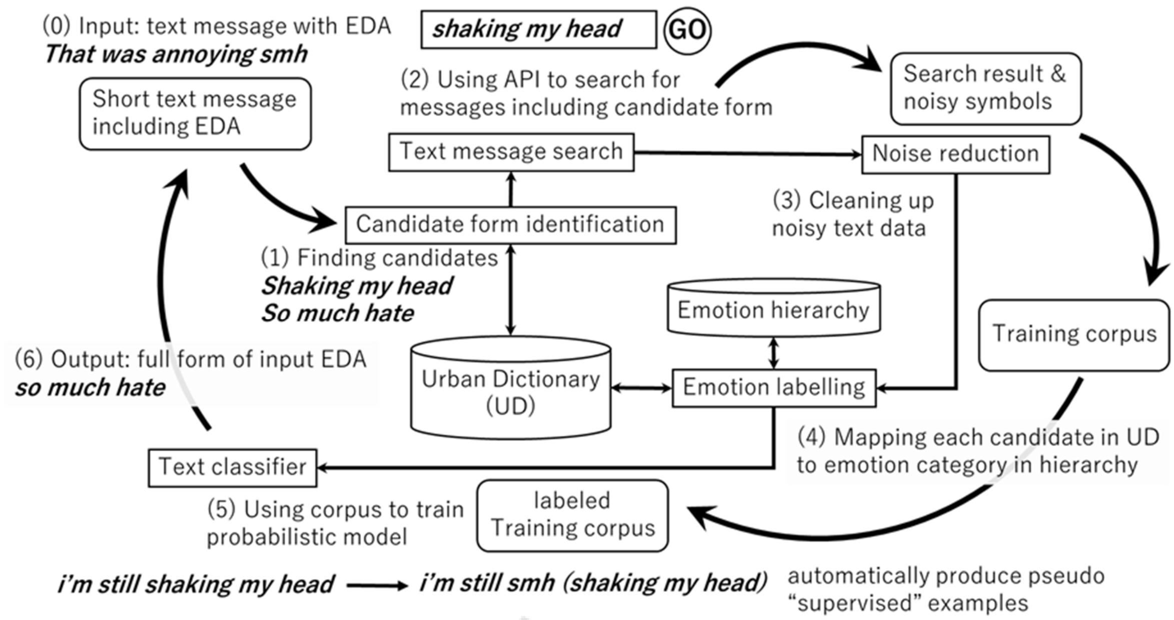

Figure 1: An overview of our method for acronym disambiguation.

Figure 1 depicts the implementations for our disambiguation method. In Figure 1, three out of five functions, represented by each rectangle, correspond to the three major factors mentioned above.

However, because as with general WSD tasks, our disambiguation task is not standalone, the structure of our system can be different from Figure 1. For example, there is a trade-off between the effectiveness, such as precision and recall, and the efficiency, such as time and space complexity. We currently put a high priority for the effectiveness because the effect of EDA is obviously associated with the quality of matching query and one of the candidates. The disambiguation does not start until the system is queried by a user, so that we can always search the latest Web for the candidates of a target acronym, sacrificing the response time. A solution is periodically collect the information from the Web to maintain our index as latest as possible, but our system can do nothing for a large number of OOV acronyms because it would be prohibited if we prepare all possible acronyms in advance. Anyway, we can use Figure 1 to explain the essence of our system, without loss of generality.

First, given a short message in which a target acronym is indicated by a user, our system searches the Web for the candidates of the target acronym. Here, each candidate is a possible full form for the acronym in question. We use a dictionary for acronym and collect a candidate list for the target acronym and a short description for each of the candidates from the dictionary. In the current implementation, we experimentally use the Urban Dictionary (https://www.urbandictionary.com), which is elaborated on Section 3.1

Second, we model each candidate that would help us select the most plausible candidate. To collect sentences that contains the target acronym used as one of candidates, such as "smh (so much hate)" or "smh (shaking my head)", we use the Twitter API to search for tweets that contains the candidate phrase, such as "so much hate", and purposefully replace the candidate phrase in each tweet with the target acronym. In practice, instead of discarding the original full phrase, we annotate this information with the sentence as meta data. By means of this, we can automatically collect a set of tweets each of which contains the target acronym annotated with the correct answer and words occurring in each tweet. At this moment in principle we can use our corpus to model the candidates. However, we intend to improve our model through emotion-bearing words.

For this purpose, we use a systematized emotional knowledge base (Shaver, Schwartz, Kirson and O'Connor, 2001), in which an emotion descriptive vocabulary is organized in a hierarchical structure as in Table 1. In Table 1, the leftmost column represents the top layer, where six types of emotions: Love, Joy,

Surprise, Anger, Sadness, and Fear are defined, and as proceeding to the right, which is equivalent to moving to a lower layer in the hierarchy, the distinction becomes finer-grained and the number of

Table 1: Shaver's emotion categories.

Primary emotion	Secondary emotion	Tertiary emotions
Love	Affection	Adoration, affection, love, fondness, liking, attraction, caring, tenderness, compassion, sentimentality
	Lust	Arousal, desire, lust, passion, infatuation
	Longing	Longing
Joy	Cheerfulness	Amusement, bliss, cheerfulness, gaiety, glee, jolliness, joviality, joy, delight, enjoyment, gladness, happiness, jubilation, elation, satisfaction, ecstasy, euphoria
	Zest	Enthusiasm, zeal, zest, excitement, thrill, exhilaration
	Contentment	Contentment, pleasure
	Pride	Pride, triumph
	Optimism	Eagerness, hope, optimism
	Enthrallment	Enthrallment, rapture
	Relief	Relief
Surprise	Surprise	Amazement, surprise, astonishment
Anger	Irritation	Aggravation, irritation, agitation, annoyance, grouchiness, grumpiness
	Exasperation	Exasperation, frustration
	Rage	Anger, rage, outrage, fury, wrath, hostility, ferocity, bitterness, hate, loathing, scorn, spite, vengefulness, dislike, resentment
	Disgust	Disgust, revulsion, contempt
	Envy	Envy, jealousy
	Torment	Torment
Sadness	Suffering	Agony, suffering, hurt, anguish
	Sadness	Depression, despair, hopelessness, gloom, glumness, sadness, unhappiness, grief, sorrow, woe, misery, melancholy
	Disappointment	Dismay, disappointment, displeasure
	Shame	Guilt, shame, regret, remorse
	Neglect	Alienation, isolation, neglect, loneliness, rejection, homesickness, defeat, dejection, insecurity, embarrassment, humiliation, insult
	Sympathy	Pity, sympathy
Fear	Horror	Alarm, shock, fear, fright, horror, terror, panic, hysteria, mortification
	Nervousness	Anxiety, nervousness, tenseness, uneasiness, apprehension, worry, distress, dread

member words generally becomes larger. Additionally, to the above mentioned six emotion categories we add "thankfulness" from the paper by (Wang, Chen, Thirunarayan and Sheth, 2012). The idea is, if we can somehow map each of the candidate for the target acronym to one of the six emotional categories in Table 1, it could be a substantially important clues to indicate the emotion of the author behind their tweet. The following two text fragments are the definitions of "smh" in UD.

- "shaking my head", smh is typically used when something is obvious, plain old stupid, or disappointment
- smh really means "so much hate". Omg she's so annoying ugh smh

As seen above, each of the underlined words is identical to that in the Sadness and Anger in Table 1, as in Sadness – Disappointment – disappointment and Anger – Irritation – annoyance. In practice, for each tweet in our training corpus, we select such emotion category that shares the maximum number of words in its subsuming words, including the category label.

After labeling all the messages with an emotion label, our dataset is used to train our candidate model. For this task, we use a probabilistic model.

The probabilistic model is often used in text classification due to its speed and simplicity. It makes the assumption that words are generated, irrespective of the position. A probabilistic model is used for our purposes to estimate the probability that a tweet belongs for a specific class. For the given EDA, we denote a class for each of its candidate. For example, each candidate of the given EDA is associated with its own prescribed class. Let's say, "shaking my head" class or "so much hate" class. For a given set of classes, it estimates the probability of a class C for the text t, with words $F = f_1 f_2 \dots f_n$and an acronym A, as in Equation (1):

$$C_k = arg \max_C P(C_k|t) \tag{1}$$

Acronyms are usually have left and right context words in text message. As we use emotion labels predicted from the definitions as an identifier for the given EDAs expansion candidate, we insert an emotion label on each of the left and right side of EDA token and denote it as E, so we modify the above equation (1) as following,

$$P(C_k|t) = \frac{P(C_k)}{P(t)} \prod_{i=1} P\left(f_i \middle| f_{i+1,\dots} EAE, f_n, C_k\right) \tag{2}$$

$P(C_k)$ and $P\left(f_i \middle| f_{i+1,\dots} EAE, f_n, C_k\right)$ are obtained through the maximum likelihood estimates (MLE). The classifier then returns the class with the highest probability in response to the submitted text message.

In Sections 3.1-3.4, we elaborate on each component in Figure 1, respectively.

3.1 Dictionary for Candidate Form Identification

To identify the possible candidate forms for the target acronym, we experimentally use Urban Dictionary (UD), which provides a large number of unofficial words and phrases, such as slangs. In November 2014, UD reached on average 72 million impressions and 18 million unique readers. The one of the unique features of UD is that visitors may agree or disagree with each definition for the given word by an up/down voting system. As there could be many unrelated definitions for the given EDA, we used that feature to choose top definitions of the ambiguous EDA to build its candidate list. For example, the number of votes (up/down) for each candidate form of "smh" was "shaking my head" (25759/ ups and 12021) and "so much hate" (970/802).

3.2 Text Message Search

In this research, we use Twitter short text messages for our training and test datasets. We use existing dataset from (Wang, Chen, Thirunarayan and Sheth, 2012) paper, where they propose automatically emotion identification approach using emotion #hashtags. In their work tweets are automatically labeled with the following seven emotions: anger, fear, joy, love, sadness, surprise and thankfulness. Some examples from their corpus are shown below.

- ZOMG user is back on glee I love him so much # love
- GN if I forget #love
- Idk why i even answered #sadness
- Omg that tickle in your throat #anger
- my sheets never stay on my bed smh #anger

Dataset contains 248898 emotion labelled tweets for training and 250000 tweets for test. Additionally, we increased dataset with 21052 emotional tweets from (Mohammad, 2012) research paper. However, not all messages in the above-mentioned corpus were available due to the removal of the messages by the users themselves. From the 1040 corrupted text messages from the above-mentioned corpuses, for our research purpose we could use only 111 tweet messages. To deal with our small side dataset, additionally to the corpuses we mentioned above, we decided to build our own emotion labelled dataset to complete our research task.

In this research, we propose our own method for data collection. After building candidates list, we query Twitter Search API with each candidate for the given EDA to collect example tweets and to build dataset for training and test. Thus, we can avoid manual labour work of EDA sense identification. For example, considering ambiguous EDA "smh", according to the UD, it has two expansion candidates:

- Shaking my head
- So much hate

To collect example messages for "so much hate", we have queried Twitter Search API as following:

- Query with full form ("so much hate")
- Query with #somuchhate
- Query with EDA + IW (identifying word or keyword)

IW here, is the word extracted from the UD definition for the given EDA. For "so much hate" definition, we found IW like: hate, annoying, irritating and etc.

Totally, we were able to collect 5067 tweets. To assure smooth training with our probabilistic model, we further pre-process our dataset.

3.3 Noise Reduction

Before training our model, we pre-process tweets by removing all non-informative:

- Emoticons ("-.-", "-_-", "^_^")
- Digital Emoticons
- Punctuation
- URLs
- #hashtags

- All doubled tweets and re-tweets
- Emotions identifier as wow, awww, xxx (“many kisses”) or kkkkk (giggling) and laugher as hahaha, hehehe, jajaja and ahahaha
- all official acronyms (“USA”, “Nasa”, “MBA”)

Numbers are substituted with their alphabetical spelling. Non-informative Twitter usernames are substituted with “user”, and all location, company (“Microsoft”, “Apple”), brand (“Adidas”, “Nike”) names are also substituted with “location”, “company” and “brand” names, respectively.

After pre-processing our data, now we have collection of Twitter text messages and need to identify/extract EDA. For the correctly identification and extraction, we build corpus consisting of officially written text documents, as an Open American National Corpus (OANC). OANC consists roughly of 15-million-word subset and it is a big electronic collection of American English. The OANC corpus includes texts documents of all genres produced since 1990.

Additionally, to the OANC corpus, we use corpora of the misspelled words from (Roger, 2009) research. His corpora consist of the 47627 words.

Our EDA extraction corpus built from the text documents collected from the above-mentioned corpora, extracts unseen tokens in their text documents from our dataset. Below are the pre-conditions for the dataset to be trained by our model:

- Text message should not be shorter than two words
- Each text message should consist at least one EDA

After pre-processing, now we have 1173 tweets to conduct our experiment.

3.4 Emotion Labelling

In this step, we use emotion descriptive words in candidate’s definition (and in some cases, when there is not enough information, we use provided examples), collected from the UD to identify its emotion label. For example:

- "shaking my head", smh is typically used when something is obvious, plain old stupid, or disappointment

After prediction, we map each emotion descriptive word with the emotion category described and categorized in Table 1 and label each message with its emotion label.

4 EXPERIMENTS

We evaluate the effectiveness of our proposed method by conducting experiments on Twitter short text messages with and without using emotion labels. We developed our own Twitter message test set containing 1173 automatically collected and pre-processed tweets using Twitter Search API for data query and UD to build candidate list. To avoid high costly and time consuming manual annotation of EDA, first, using UD look up method, we build candidate list for each EDA. Second, we queried Twitter API using each candidate for the given EDA. Totally 5067 messages were collected. After performing pre-processing steps on collected data to remove all the noises, we were left with 1173 short text messages, 10 unique EDA with 21 candidates in total.

For the experiment purposes, candidates were replaced with their EDA, respectively and labelled with its emotion category (anger, fear, joy, love, sadness, surprise and thankfulness). To identify emotional state for each candidate of the given EDA, we use emotion descriptive words from candidate’s dictionary definition. For example, following is the top definitions for the two candidates of “smh”:

- meaning, "shaking my head", smh is typically used when something is obvious, plain old stupid, or disappointment
- smh really means "so much hate". Omg she's so annoying ugh smh

By using these definitions for each candidate of the given EDA, we choose emotional state from seven emotion categories described in Table 1, where emotions were categorized into a short tree structure.

We compare our system by conducting experiment on datasets – dataset labelled with emotion label and dataset without any emotion label. Each dataset consists of 946 tweets for the training of our probabilistic model and 227 tweets for testing. As we suspect, that we might have some imbalance due to some classes having more examples compare to others, we estimate micro-average for precision and recall for each EDA (Table 2), which is preferable in multi-classification setup. Table 3 shows our experiment results per classes, which was calculated using confusion matrix for each class.

To validate our results, we performed 3-fold cross validation on our dataset.

From the results by the comparison of precision and recall for each class between our proposed method and baseline, we can observe, that our proposed disambiguation method improves baseline results.

Table 2: Effectiveness of acronym disambiguation for different methods per EDA.

EDA	Precision		Recall		F1 score	
	Baseline	Our method	Baseline	Our method	Baseline	Our method
bae	0.53	0.54	0.91	1.00	0.66	0.70
bf	0.52	0.73	0.53	0.67	0.51	0.68
fbf	0.45	0.48	0.82	0.88	0.58	0.61
hth	0.38	0.44	0.81	0.75	0.48	0.54
ily	0.83	0.88	0.68	0.82	0.73	0.84
kiss	0.67	0.88	0.44	0.68	0.53	0.76
lol	0.71	0.84	0.73	0.65	0.72	0.72
lig	0.40	0.52	0.81	0.96	0.53	0.67
smh	0.56	0.99	0.39	0.79	0.45	0.88
tftf	0.00	0.17	0.00	0.00	0.00	0.22
Overall	**0.50**	**0.65**	**0.61**	**0.72**	**0.52**	**0.66**

Table 3: Effectiveness of acronym disambiguation for different methods per candidate.

Candidate	Precision		Recall		F1 score	
	Baseline	Our method	Baseline	Our method	Baseline	Our method
babe	0.62	0.64	0.91	1.00	0.73	0.77
best_at_everything	0.00	0.00	0.00	0.00	0.00	0.00
best_friend	0.63	0.88	0.68	0.64	0.52	0.69
boyfriend	0.46	0.52	0.62	0.84	0.47	0.63
facebook_flirting	0.00	0.00	0.00	0.00	0.00	0.00
female_best_friend	0.00	0.00	0.00	0.00	0.00	0.00
flash_back_friday	0.71	0.77	0.82	0.88	0.75	0.80
happy_to_help	0.50	0.53	0.96	0.90	0.63	0.65
hope_that_helps	0.00	0.00	0.00	0.00	0.15	0.28
i_love_you	0.92	0.97	0.68	0.82	0.76	0.88
im_leaving_you	0.00	0.00	0.00	0.00	0.00	0.00
keep_it_simple_stupid	0.00	0.67	0.00	1.00	0.12	0.77
kiss	0.97	0.98	0.43	0.62	0.60	0.76
laughing_out_loud	0.82	0.98	0.72	0.60	0.74	0.72
let_it_go	0.50	0.60	0.76	0.93	0.59	0.72
life_is_good	0.30	0.43	1.00	1.00	0.44	0.59
lots_of_love	0.57	0.63	0.81	0.93	0.65	0.74
shaking_my_head	0.32	0.98	0.28	0.72	0.30	0.82
so_much_hate	0.74	1.00	0.44	0.87	0.55	0.93
thanks_for_the_fellow	0.00	0.56	0.00	0.67	0.00	0.50
too_funny_to_fix	0.00	0.00	0.00	0.00	0.00	0.00

We could improve high precision score of 0.65. It shows, that our proposed method is applicable for EDA disambiguation task. Low precision and recall score is due to the small size of the given EDA. For some EDA we could not collect many example messages during the data collection step. Some candidates of the given EDA were too long, so when we queried Twitter API with it we could retrieve only small size of data. We assume, that it is due to the length of the candidate, as some candidates ("thanks_for_the_follow") are too long too type, so users prefer to type its shortened version ("tftf").

Also, P, R and F-score for the "tftf" acronym was lower when candidates are more semantically similar. "tftf" has two candidates "thanks for the follow" and "too funny to fix". We could identify identical emotion label, as "joy" for both candidates. In this situation, context words around the EDA could be very helpful.

For the future work, we would like to improve the performance of our system by considering disambiguation of EDA's with several emotion labels, like "bf" could have more than one emotional state (anger, fear, joy, love, sadness, surprise and thankfulness) according to the context of the message.

5 CONCLUSIONS

In this paper, we have proposed an approach for disambiguation of emotion-driven acronyms (EDA) in short text messages by using its emotion descriptive words evoked from the message.

Our proposed method generates candidates list for the given EDA by looking up Urban Dictionary and query Twitter Search API with candidates full from to automatically collect data for training and testing. For data collection, we have proposed new approach to collect the data without any manually annotation. We proposed a method, which collects the data by querying Twitter Search API with candidate of the given EDA plus EDA indicator word, which is extracted from the EDA definition on UD webpage. Thus, we do not need to manually annotate Twitter text messages.

For the identification of emotional state for the given EDA, we also used description provided by Urban Dictionary and using seven emotion categories from the existing studies, we could automatically identify emotion label for the given EDA.

In disambiguation task, we conducted two experiments using emotion labelled dataset and dataset without any emotion label for the performance evaluation. We could achieve high F-score by integrating dictionary lookup, automatically collecting and labelling and probabilistic LM.

In future work, we plan to improve the performance of our system by considering EDA with many emotion labels and EDA with identical emotion labels.

REFERENCES

Barua, J., Patel, D., 2016. Discovery, Enrichment and Disambiguation of Acronyms. Springer International Publishing, LNCS, vol. 9829, pp. 345-360.

Boguraev, B.K., Chu-Carroll, J., Ferrucci, D.A., Levas, A.T., and Prager, J.M., 2015. Context-Based Disambiguation of Acronyms and Abbreviations. United States Patent. Patent No.: US 9,020,805 B2.

Bracewell, D.B., Russell, S., and Wu, A.S., 2005. Identification, Expansion and Disambiguation of Acronyms in Biomedical Texts. In ISPA Workshops, LNCS, vol. 3759, pp. 186-195.

Charbonnier, J., Wartena, C., 2018. Using Word Embeddings for Unsupervised Acronym Disambiguation. In 27th International Conference on Computational Linguistics, pp. 2610-2619. New Mexico, USA

Duque, A., Martinez-Romo, J., Araujo, L., 2016. Can multilinguality improve Biomedical Word Sense Disambiguation. Journal of Biomedical Informatics, vol. 64, pp. 320-332.

Sridhar, V.K.R., 2015. Unsupervised Text Normalization Using Distributed Representations of Words and Phrases. In NAACL-HLT, pp. 8-16. Denver, Colorado.

Li, C., Ji, L., and Yan, J., 2015. Acronym Disambiguation Using Word Embedding. In 29th of AAAI Conference on Artificial Intelligence, pp. 4178-4179. Austin, TX.

Mohammad, S.M., 2012. #Emotional Tweets. In 1st of Joint Conference on Lexical and Computational Semantics, pp. 246-255. Montreal, Canada.

Moon, S., McInnes, B., and Melton, G.B., 2015. Challenges and Practical Approaches with Word Sense Disambiguation of Acronyms and Abbreviations in the Clinical Domain. Healthcare Informatics Research, 21(1):35-42.

Roger, M., 2009. Ordering the Suggestions of a Spellchecker Without Using Context. Natural Language Engineering, 15(2):173-192.

Shaver, P., Schwartz, J., Kirson, D., and O'Connor, C., 2001. Emotional Knowledge: Further Exploration of a Prototype Approach. In: G. Parrott (Eds.), Emotions in Social Psychology: Essential Readings, 26-56. Philadelphia, PA.

Sproat, R., Black, A., Chen, S., Kumar, S., Ostendorf, M., and Richards, C., 2001. Normalization of Non-Standard Words. Computer Speech and Language, 15(3): 287-333.

Wang, W., Chen, L., Thirunarayan, K., and Sheth, A.P., 2012. Harnessing Twitter "Big Data" for Automatic Emotion Identification. In: 4th IEEE Conference on Social Computing, pp. 587-592. Washington, DC.

Xue, Z., Yin, D., and Davison, B.D., 2011. Normalizing Microtext. In: AAAI-11: Workshop on Analyzing Microtext, pp. 74-79. San Francisco, CA.