Natural Logic Inference for Emotion Detection

Han Ren[1], Yafeng Ren[2†], Xia Li[1], Wenhe Feng[1] and Maofu Liu[3]

[1] Laboratory of Language Engineering and Computing, Guangdong University of Foreign Studies, Guangzhou 510420, China
[2] Collaborative Innovation Center for Language Research and Services, Guangdong University of Foreign Studies, Guangzhou 510420, China
[3] College of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, Hubei 430065, China
hanren@gdufs.edu.cn, {201610151, 200211025, 201610128}@oamail.gdufs.edu.cn, liumaofu@wust.edu.cn

Abstract. Current research on emotion detection focuses on the recognizing explicit emotion expressions in text. In this paper, we propose an approach based on textual inference to detect implicit emotion expressions, that is, to capture emotion detection as an logical inference issue. The approach builds a natural logic system, in which emotional detection are decomposed into a series of logical inference process. The system also employ inference knowledge from textural inference resources for reasoning complex expressions in emotional texts. Experimental results show the efficiency in detecting implicit emotional expressions.

Keywords: Natural Logic, Textual Inference, Emotion Detection, Implicit Emotional Expression.

1 Introduction

Emotion detection refers to the identification of emotional expressions in texts, which are, although there is still no strict definition, generally categorized as happiness, sadness, anger, shame, and so on. As one of the most important research topics in natural language processing, emotion detection is widely used in opinion mining, product recommendation, dialog system, and so on[1].

Current research on emotion detection can be mainly classified as the feature-based and knowledge-based approaches. Feature-based approaches employ machine learning algorithms to classify emotion by building appropriate features[2, 3, 4], while knowledge-based approaches employ emotion lexicons[5], domain lexicons[6] or patterns extracted[7] to detect emotions. Many approaches show state-of-the-art performances in detecting explicit emotion expressions. However, it is hard to achieve acceptable performances of detecting implicit emotion expressions. For example,

T*: The husband breaks his wife's head*.

† Corresponding Author

Since there is no explicit emotion words in the sentence, it is difficult to identify the emotion of the agent *husband* by emotion lexicons or to define appropriate features to detect the emotion. In fact, *break* entails the action *hit*, while *hit* holds the emotion of *anger*[1]. Apparently, such inference process contributes to identifying the emotion and its holder.

Based on this idea, we propose an approach, that is, to treat emotion detection as an textual inference problem. Textual inference refers to that, given a text fragment, the goal of textual inference is to recognize a hypothesis that can be inferred from it or not. Textual inference is a notable field of research that are leveraged in many natural language processing areas, such as document summarization, information retrieval and question answering[8].

For emotion detection, a series of premises derived from the emotional sentence T can be generated, for example, a premise H can be *The husband is angry*. Then we judge if H can be inferred from T. If so, it means that the meaning of H is contained in T, in other words, H is true. Therefore, we can draw a conclusion that the emotion of the husband is angry. The idea has two advantages: 1)an implicit emotion detection problem can be formalized as an textual inference one, which can be handled by many state-of-the-art textual inference models for better performances; 2)the emotion holder in a sentence can be easily identified if a premise generated by such sentence is judged to be true.

In this paper, a natural logic-based approach is proposed to model the emotion detection process, and the aim is to capture emotional inference by appealing directly to the logical expressions. In this approach, inference relations between texts are viewed as logical containment ones[9], and emotional reasoning between semantic units in a same synset can be carried out like: $break \sqsubseteq hit$, $hit \sqsubseteq angry$, which shows a distinct cue of implicit emotions. Also, inferable relations without emotions are identified in this approach, like other textual inference systems, to benefit the emotional reasoning. Experiments show that the approach shows a higher efficiency on detecting implicit emotion expressions.

The rest of this paper is organized as follows. Section 2 gives a brief introduction of related work. Section 3 and 4 gives a detailed description of the emotion detection approach based on natural logic. Section 5 shows experimental results as well as some discussions. Finally, the conclusion is drawn in Section 6.

2 Related Work

Knowledge-based Approaches for emotion detection focus on the construction of affective lexicons, and the assumption is that rich knowledge with emotion expressions will help to detect explicit emotion as well as implicit one. The most well-known emotion lexicon in English is WordNet-Affect, which annotates the words that represents emotional concepts in WordNet, then expands emotion word lists through synsets that contains those words. Based on the idea, emotion lexicons of other languages such as Japanese[10] and Chinese[5] are built. However, such approaches may

[1] The mapping relation can be found in emotion lexicons, such as EmoLex.

achieve low performances when facing complex emotion expressions, since the emotion of a single word can not determine the overall emotion type of expressions.

In order to handle emotion detection without complex knowledge machining, machine learning-based approaches are proposed, and feature engineering becomes a main strategy in order to profile emotional expressions in multiple aspects. Rao et al.[2] used spectral and prosodic features to recognize emotions, while Xu et al.[4] employed intra-sentences features as well as sentential contexts to classify emotions for sentential emotion expressions. In order to acquire features effectively and automatically, deep learning-based approaches are proposed, and researchers employ various deep models such as CNN[11], RNN[12] and LSTM[13] to identify sentiment in social texts. Although such approaches achieve better performances than lexicon-based or rule-based approaches in most cases, it is still difficult to detect implicit emotions in texts, since finding implicit emotions needs such systems having the ability of complex knowledge inference or deep semantic analysis.

3 Natural Logic for Emotion Detection

Logic-based approaches have been explored for textual inference, and the aim is to capture logical inferences by appealing directly to the structure of language. In comparison with other logic systems, natural logic is more appropriate to handle inference relations in natural language, since it has no need to define complex and strict production rules. It is also appropriate to handle commonplace phenomena such as negation, antonymy and downward-monotone quantifiers[14], which usually appear in emotional texts.

Based on the theoretical framework on natural logic and monotonicity calculus[15, 16], natural logic systems such as NatLog[17] and NaturalLI[18] are built for textual inference. Inspired by them, we construct a complete proof system in three parts: 1)define atomic relations; 2)define monotonicity over atomic relations and; 3)describe a proof system. Different from those systems, our system takes emotional relations into account, which provide efficiency for reasoning emotional expressions.

3.1 Atomic Relations

Atomic relations represent logical relations between fundamental units for logic deduction. In terms of textual inference, the uppermost relation is entailment, that is, one text entails another or not. In natural logic, entailment is viewed as a semantic containment relation, which can be analogous to containment relation in set relations. For example, the concept *car* entails the concept *vehicle* means the set *vechicle* contains the concept *car*. Following this idea, relations referring inference are easy to be defined using set relations.

Another important relation in the system is affection, which shows a relation between a state or action and an emotion reflected by it. Sometimes, affection can be roughly viewed as entailment, for example, the action *cry* reflects the emotion *sadness*, and it can be explained that *cry* entails *sadness*. In fact, however, affection expresses emotional reflection rather than containment relation in a fine view. For example, *celebrate* may indicates the emotion *joy*, whereas it is inappropriate to say that *celebration* belongs to the set *joy*. For a better understanding, a specialized relation should be defined to denote the relation between a state or action and an emotion reflected.

Table 1. Atomic relations in the system.

relation	description
$x \sqsubseteq y$	$x \to y, y \nrightarrow x$
$x \sqsupseteq y$	$x \nrightarrow y, y \to x$
$x \equiv y$	$x \to y, y \to x$
$x \wedge y$	x is the negation of y
$x \triangleright y$	x reflects y
$x \# y$	x and y are irrelative

Following the idea in [14], we define six relations shown in Table1. In the table, relation represents logical relations, while description means semantic inference ones, in which $x \to y$ denotes x entails y and $x \nrightarrow y$ denotes x does not entail y.

Some examples for each atomic relation are shown as follows. Note that a denotation may be a lexiconsl entry, a phrase or an entailment rule, for example, $help\ X \sqsubseteq give\ X\ a\ hand.$

$$\begin{aligned} car &\sqsubseteq vehicle \\ plane &\sqsupseteq jet\ plane \\ U.S. &\equiv America \\ agree &\curlywedge not\ agree \\ celebrate &\triangleright joy \\ car &\# joy \end{aligned}$$

For the validity of emotional inference, some constraints are defined: in the first relation $\sqsubseteq$ in Table 1, if x is an emotional denotation, y must be an unemotional one; in the second relation $\sqsupseteq$, if y is an emotional denotation, x must be an unemotional one; in the third and the fourth relation, denotations of both side have the same category, namely they are both emotional denotations or unemotional ones; in the relation of affection $\triangleright$, x must be an unemotional denotation and y must be an emotional one. Such constraints ensure the monotonicity of emotional reasoning, namely each reasoning step referring emotions always proceeds from factive denotations to emotional ones.

3.2 Monotonicity

Monotonicity describes the containment relation between a meaning and its extension and intension of a denotation. More specifically, monotonicity shows if a meaning can be inferred from any premise set containing or contained by a set, i.e., subset or superset, which is the premise set that the meaning can be inferred from it. For textual inference, the aim of monotonicity calculus it to map semantic relations of text pieces(words, phrases, etc.) to semantic relations of sentences.

There are, generally, three monotonicity classes, that is, upward-monotone, downward-monotone and non-monotone. The upward-monotone indicates the extension of a meaning, while the downward-monotone indicates the intension of a meaning. In textual inference, many entailment phenomena can be cast as upward-monotone functions, such as *buy a car* $\sqsubseteq$ *buy a vehicle*, since *car* $\sqsubseteq$ *vehicle*. As to language expressions including negative words, downward-monotone may exists, such as *have not a vehicle* $\sqsubseteq$ *have not a car*, since *car* $\sqsubseteq$ *vehicle*. Table 2 shows the mapping relations for downward-monotone in the system.

Table 2. Downward-monotone mapping relations. *r* is the relation between two text pieces(e.g., words), and f_{-} is the relation of the two sentences except the relation *r*.

r	$\sqsubseteq$	$\sqsupseteq$	$\equiv$	$\wedge$	$\rhd$	$\#$
f	$\sqsupseteq$	$\sqsubseteq$	$\equiv$	$\wedge$	$\#$	$\#$

As an example, a formal description of monotonicity over three classes for the relation entailment is shown as follows, along with MacCartney and Manning [17]: for all $x, y \in D$, the function *f* is upward-monotone iff $x \sqsubseteq y$ entails $f(x) \sqsubseteq f(y)$, *f* is downward-monotone iff $x \sqsubseteq y$ entails $f(y) \sqsubseteq f(x)$, and *f* is non-monotone iff $x \sqsubseteq y$ neither entails $f(x) \sqsubseteq f(y)$ nor $f(y) \sqsubseteq f(x)$.

There is still an issue when reasoning emotional expressions, that is, how to define monotonicity for the relation of affection. Since such relation can be roughly treated as the relation entailment, we may define it with an upward-monotone function, that is, for all $x \in D$, $y \in M$, the function *f* is upward-monotone iff $x \rhd y$ entails $f(x) \rhd f(y)$. Here *D* denotes an unemotional expression set and *M* an emotional expression set. However, such monotonicity calculus may lead to wrong inference. For example, according to the relation *beat* $\rhd$ *angry*, we may obtain a correct relation that *The husband beat his wife* $\rhd$ *The husband is angry* and a wrong relation that *The husband beat his wife* $\rhd$ *His wife is angry*. The reason lies in that the relation *beat* $\rhd$ *angry* holds only if the agent of the action beat is same with the holder of the emotion *angry*. More specifically, an action may lead to multiple emotions, thus yielding an appropriate emotional reasoning depends on judging if the agent or patient of such action and the holder of an emotion is same.

Therefore, it is necessary to make constraints for the monotonicity function of the relation affection. We re-define an upward-monotone function for the relation affection as follows: the function *f* is upward-monotone iff $x \rhd y$ entails $f(x) \rhd f(y)$, and the agent of x and the holder of the emotion y are same. Here we employ Stanford parser[2] to acquire agent of each sentence by shallow semantic analysis. If such agent is not found, the emotion holder labeled in the sentence will be the default agent.

[2] http://nlp.stanford.edu/software/lex-parser.shtml

3.3 Proof System

The aim of the proof system is to join semantic relations, by which semantic relations between sentences can be inferred according to all atomic relations. Inferences iteratively join two relations together to get the final entailment relation. Such join relations are shown in Table 3.

Table 3. The join table of the proof system. Note that join relation is also subject to constraints mentioned in section 3.1. For example, for $x \rhd y,\ y \sqsubseteq z$, if y is an emotional denotation and z is an unemotional denotation, then $\rhd \bowtie \sqsubseteq = \#$.

$\bowtie$	$\sqsubseteq$	$\sqsupseteq$	$\equiv$	$\wedge$	$\rhd$	$\#$
$\sqsubseteq$	$\sqsubseteq$	$\#$	$\sqsubseteq$	$\sqsupseteq$	$\rhd$	$\#$
$\sqsupseteq$	$\#$	$\sqsupseteq$	$\sqsupseteq$	$\sqsubseteq$	$\#$	$\#$
$\equiv$	$\sqsubseteq$	$\sqsupseteq$	$\equiv$	$\wedge$	$\rhd$	$\#$
$\wedge$	$\#$	$\#$	$\wedge$	$\equiv$	$\#$	$\#$
$\rhd$	$\rhd$	$\#$	$\rhd$	$\#$	$\#$	$\#$
$\#$	$\#$	$\#$	$\#$	$\#$	$\#$	$\#$

The following example illustrates the proof process using the join table. Considering the text pair(T,H):

T: *The husband breaks his wife's head*.
H: *The husband is angry*.

Relations between T and H are listed as follows:

r_1: $\sqsubseteq$ (*break*, *hit*)
r_2: $\sqsubseteq$ (*his wife's head*, *his wife*)
r_3: $\rhd$ (*hit*, *angry*)

Assume $S = \{s_i\}$ is a transforming text set, let $s_0 = T,$ an inference $T \rightarrow H$ proceeds by iteratively joining two relations like $s_{i+1} = s_i \bowtie r_i$ until the last output $s_n = H$. Note that if there is any downward-monotone expression, such as negative word, in *T*, we should use $f(r_i)$ instead of r_i to join with s_i.

4 Natural Logic Inference

Natural logic inference can be cast as a search problem, that is, given a query T, we search possible facts for a valid text H over the space. Angeli and Manning[18] introduced a natural logic based search approach by building a graph, in which nodes are text pieces derived from the query and the edges describe mutation of these text pieces. Transitions along the search denote inference steps in natural logic. Following the idea, we build a graph of transition candidates with a learning algorithm for emotional inference. Different with their approach, our graph builds nodes with emotional text pieces, that is, we builds transition candidates as nodes from T as well as emotional text pieces as nodes from H; we adopt multiple inference resources such as inference rules rather than only lexiconsl knowledge from WordNet for transition; we also introduce a method to estimate the contribution of each resource by parameter learning.

4.1 Inference Graph

The space of possible nodes in the search is the set of possible partial derivations. Each node constructed is a pair (*t*, *s*), in which *t* is a partial derivation from T and *s* is the state in the state set $\{valid, invalid\}$. Each edge expresses a mutation of a single text piece, such as a lexiconsl replacement or an entailment rule based transition.

Node Considering that each T and automatic-generated H probably intuitively has a very different meaning, we generate nodes that approach the meaning of H as far as possible by using SentiSense[3]. a concept-based affective lexicon. The lexicon provides a mapping between emotional expressions and WordNet synsets, that is, assign a meaning of a word with a corresponding emotion that the word holds. The process of node generation is describes as follows: first, verbs, nouns and adjectives appeared in T are handled with NLTK[4], a natural language toolkit, for word sense disambiguation; then, find synsets or hypo-synsets of those words that having the same emotions with H. Finally, nodes are generated by using other words in the same synset to replace the original word in T and connect with the node of T. We also use synonyms of emotional word in H to generate nodes adjoining the node of H. All these nodes built are labeled valid without the need of logic proof. The advantage of generating adjoining nodes to T and H is to narrow the difference of both semantic gap and formal expression between T and H so that search can be proceeded smoothly.

Edge An edge describes a transition of a text piece in one node to an entailment text in another. As illustrated in [18], a mapping is built to bridge type of edges(synonym, delete word, etc.) and atomic relations in Natural Logic. According to this idea, a graph search process can be cast as a logic proof one. We adopt such method to build the inference process in the our system.

To generate edges, the approach in [18] employ lexiconsl relations in WordNet and some edit changes to describe transitions. However, many entailment transitions are beyond lexiconsl relation. For example, *X help Y* $\rightarrow$ *X give Y a hand*. Such variation illustrates a paraphrase relation, which is a semantic combination of lexiconsl relations and syntactic variation. In the system, we employ multiple entailment knowledge resources to express transitions between nodes, in order to achieve a better transition performance. Since atomic relations in the system are also able to describe relations between text pieces such as phrases and entailment rules, as shown in Section 3.1, they can also illustrate the transition among nodes. Table 4 shows each type of edges and its corresponding atomic relation.

Table 4. Types of edges and corresponding atomic relations. DIRT, binaryDIRT and MRPC are paraphrase collections, TEASE and FRED are entailment rule collections, , WikiRules! is a

[3] http://nlp.uned.es/~jcalbornoz/SentiSense.html
[4] http://www.nltk.org/

collection of lexicons reference relation in Wikipedia, and Google Distance is an semantic similarity metric viewed as a synonym resource.

Resource[5]	Type of edge	Relation
WordNet	Hypernym	$\sqsubseteq$
	Hyponym	$\sqsupseteq$
	Synonym	$\equiv$
	Antonym	$\wedge$
	Meronym	$\sqsubseteq$
DIRT	Paraphrase	$\equiv$
TEASE	Entailment	$\sqsubseteq$
FRED	Entailment	$\sqsubseteq$
WikiRules!	entailment	$\sqsubseteq$
binaryDIRT	Paraphrase	$\equiv$
MRPC	paraphrase	$\equiv$
Google Distance	co-occurrence	$\equiv$

After constructing an inference graph, textual inference is to find a nearest path from T to H over the graph leveraging on logic proof through inference resources.

4.2 Learning for Inference

Note that transitions derived from these knowledge bases are not always valid. A simple reason is that two parts of each paraphrase or entailment rule is not always semantically identical, and we cannot estimate semantic relation for each pair. Alternately, we can estimate the validity of each resource as an approximate performance for search(or inference). The proposed approach is described as follows: each resource has a feature f_i denoting all transitions using it, and a parameter θ_i is assigned to each f_i, denoting the validity degree of each resource. Then the transition validity for each inference process can be estimated through computing $\sum \theta_i f_i$. Given a score threshold α, we say that the inference is valid if

$$\sum \theta_i f_i + \alpha \geq 0 \qquad (1)$$

Here f_i can be computed by:

$$f_i = p(t' \to h) - p(t \to h) \qquad (2)$$

where $p(t \to h)$ denotes the probability of text *t* entails *h*, and $p(t' \to h)$ denotes the probability of transformed text *t'* entails h leveraging rules in the resource having feature f_i. Here we simply define *p* as the n-gram overlap. Intuitively, if the more the count of overlapped n-grams are, the more the transition is valid. Note that since auto-generated H is always an emotional expression and probably contain less words in T, it is necessary to generate *h* using affection lexicons so that the value of f_i is not trivial.

The learning process is a parameter estimation process of finding the optimal $(\hat{\theta}, \hat{\alpha})$. We apply averaged perceptron, a supervised linear learning model, to estimate parameters. For a new inference process, it is valid if the equation $\sum \hat{\theta} F + \hat{\alpha} \geq 0$ is satisfied.

[5] http://aclweb.org/aclwiki/index.php?title=Textual_Entailment_Resource_Pool

5 Experiments

5.1 Data

Our evaluation data comes from International Survey of Emotion Antecedents and Reactions(ISEAR)[6], a affection corpus with 7 major emotions(joy, fear, anger, sadness, disgust, shame and guilt). Since ISEAR is a self-report emotion dataset, most emotion holders in texts are *I*, *my* or *me*. We delete data without the *I*, *my* or *me* from the corpus and get a total 5898 items. For each item we use a simple pattern *I feel Y* to generate hypothesis automatically. Here *Y* is replaced by an adjective word in the labeled emotion class of the item. A statistical result of the evaluation dataset is shown in Table 5.

Table 5. Statistical results of the evaluation dataset.

emotion	#
joy	873
fear	859
anger	842
sadness	725
disgust	776
shame	891
guilt	932
Total	5898

We choose 4000 items in the evaluation dataset as the training data and the rest 1898 items as the test data. For each item in the test data, six hypothesis texts such as "*I feel angry*", "*I feel happy*" are built automatically.

5.2 Results

Two experiments are settled. The first one is to evaluate the system in this paper against other emotion detection systems. In the experiment, four systems are set: the first system(Lex) is a lexicon-based one, that is, we count the number of emotion words in a text for each emotion type, then the emotion label of the text is simply determined as the emotion type having most emotion words. The second system(SVM-word) is a feature-based one, that is, we employ word features to classify emotion using SVM. The second system(SVM-combined) is also a feature-based one, and the difference from the prior system is that, we employ word features as well as emotion lexicon features to build an SVM classifier; here each emotion lexicon feature is a count value of emotional words for each emotion type according to emotion lexicons. The fourth system(NaLogic) is the system proposed in this paper, that is, we use natural logic inference to detect emotions in texts. Evaluation metrics are precision, recall and F-1 score. Experimental results are shown in Table 6.

The experiment results show that:

[6] http://emotion-research.net/toolbox/toolboxdatabase.2006-10-13.2581092615

Table 6. Experiment results for emotion detection.

		joy	fear	anger	sadness	disgust	shame	guilt
Lex	P	0.4204	0.3839	0.3342	0.3009	0.3823	0.1786	0.3241
	R	0.3596	0.3448	0.2970	0.2208	0.2509	0.1917	0.2587
	F1	0.3876	0.3633	0.3145	0.2599	0.3030	0.1849	0.2877
SVM-word	P	0.4825	0.4670	0.3801	0.3714	0.4677	0.2983	0.3602
	R	0.4029	0.4035	0.3516	0.2629	0.3128	0.2544	0.2954
	F1	0.4402	0.4329	0.3653	0.3079	0.3749	0.2746	0.3246
SVM-combined	P	0.5208	0.4793	0.3972	0.3908	0.4851	0.3106	0.3737
	R	0.4476	0.4656	0.3854	0.2711	0.3198	0.2793	0.3080
	F1	0.4814	0.4724	0.3912	0.3201	0.3855	0.2941	0.3377
NaLogic	P	0.5463	0.4970	0.4202	0.4184	0.5263	0.3503	0.3881
	R	0.4626	0.5002	0.3917	0.2740	0.3385	0.2767	0.3204
	F1	0.5010	0.4986	0.4054	0.3311	0.4120	0.3092	0.3510

1)The approach in this paper outperforms lexicon-based and feature-based approaches in this experiment. In comparison with the performance of the second best system(SVM-combined), the system of natural logic inference achieves an increasing 1.95% F1 performance of emotion joy, an increasing 2.62% of fear, an increasing 1.42% of anger, an increasing 1.1% of sadness, an increasing 2.65% of disgust, an increasing 1.51% of shame and an increasing 1.33% of guilt. It indicates that the inference-based approach helps to detect emotions in comparison with lexicon-based and feature-based approaches, especially for implicit emotions.

2)It is better to adopt machine learning approaches rather than lexicon-based approaches to detect implicit emotions. In comparison with the lexicon-based system Lex, the system SVM-word achieves an increasing 5.26% F1 performance of emotion joy, an increasing 6.96% of fear, an increasing 5.08% of anger, an increasing 4.79% of sadness, an increasing 7.19% of disgust, an increasing 8.97% of shame and an increasing 3.69% of guilt, which occurs a distinguished performance increase than those between the NaLogic system and the SVM-combined system.

3)Knowledge from emotion lexicons helps to improve emotion detection performance. In comparison with the SVM-word system, the SVM-combined system achieves an increasing 4.12% F1 performance of emotion joy, an increasing 3.94% of fear, an increasing 2.59% of anger, an increasing 1.23% of sadness, an increasing 1.06% of disgust, an increasing 1.95% of shame and an increasing 1.31% of guilt. The reason is that, there are some implicit emotion expressions that cannot be detected by machine learning approaches are provided by emotion lexicons, thus using them will contribute to improving performance for detecting implicit emotions.

In addition, for detecting emotion joy, the NaLogic system achieves a best performance, while for shame, the system achieves a lower one. The reason probably lies in that there are more clue words for detecting the emotion joy than other emotions, such as *birthday* or *anniversary*, while there are less clue words for detecting the emotion shame than other emotions.

The second experiment investigates the contributions of knowledge resources listed in Table 4. The experiment is set as follows: every time only one resource is removed from the NaLogic system, and we use macro F1 score to evaluate the overall performance for detection of all emotion types. Experimental results are shown in Table 7.

Table 7. Knowledge resource evaluation results.

	macro F1
NaLogic	0.4012
-WordNet	0.3635
-DIRT	0.3895
-TEASE	0.3907
-FRED	0.3919
-WikiRules!	0.3844
-binaryDIRT	0.3956
-MRPC	0.3917
-Google distance	0.3870

It can be seen from the evaluation results that, system achieves a lowest performance by removing WordNet. The reason lies in that, WordNet provides a good many of words and synsets, which may help to build the inference graph by bridging mappings between emotion expressions and common words in cooperation with SentiSence. It can be also seen that, the systems of removing WordNet, DIRT or WikiRules! achieve more decreasing performances in comparison with the performances of other system. Since such knowledge bases have a larger data scale than others in this experiment, it indicates that the scale of knowledge resources impact the construction of the inference graph, which will eventually influence the system performances.

6 Conclusion

In this paper, we propose a novel approach, that is, to detect implicit emotions by textual inference. Following this idea, an implicit emotion detection problem can be formalized as an textual inference one, which can be handled by many textual inference models for better performances, and the emotion holder in a sentence can be easily identified according to the premise generated. To this end, we build an emotional inference system that employs natural logic to handle emotional relations as well as non-emotional ones, and the experimental results show its efficiency of reasoning expressions with emotions.

Acknowledgements

This work is supported by National Natural Science Foundation of China(61402341, 61402119) and Bidding Project of GDUFS Laboratory of Language Engineering and Computing(LEC2016ZBKT001, LEC2016ZBKT002).

References

1. Das, D. and Bandyopadhyay, S.: Emotion Analysis on Social Media: Natural Language Processing Approaches and Applications. In: Agarwal et al. (eds.) Online Collective Action: Dynamics of the Crowd in Social Media, Lecture Notes in Social Networks, pp.19-37, Springer (2014).
2. Rao, K. S. and Koolagudi, S. G.: Robust Emotion Recognition Using Spectral and Prosodic Features. New York, Springer (2013).
3. Reisenzein, R., Hudlicka, E., Dastani, M., Gratch, J., Hindriks, K., Lorini, E., et al.: Computational Modeling of Emotion: Toward Improving the Inter- and Intradisciplinary Exchange. IEEE Transactions On Affective Computing, 4(3) (2013).
4. Xu, J., Xu, R., Lu, Q. and Wang, X.: Coarse-to-Fine Sentence-Level Emotion Classification Based on The Intra-Sentence Features And Sentential Context. CIKM2012, Maui, USA (2012).
5. Xu, R., Gui, L., Xu, J., Lu, Q. and Wong, K. F.: Cross Lingual Opinion Holder Extraction based on Multiple Kernel SVMs and Transfer Learning. International Journal of World Wide Web, 18(2) (2013).
6. Andreevskaia, A. and Concordia, S. B.: Mining WordNet for a Fuzzy Sentiment: Sentiment Tag Extraction from WordNet Glosses. EACL2006 (2006).
7. Xu, R. and Wong, F. F.: Coarse-Fine Opinion Mining - WIA in NTCIR-7 MOAT Task. NTCIR-7 Workshop, Tokyo, Japan (2008).
8. Androutsopoulos, I. and Malakasiotis, P.: A Survey of Paraphrasing and Textul Entailment Methods. Journal of Artificial Intelligence Research, 38(1), 135-187 (2010).
9. MacCartney, B. and Manning, C. D.: An extended model of natural logic. Proceedings of the 8th International Conference on Computational Semantics, Tilburg, Netherland (2009).
10. Torii, Y., Das, D., Bandyopadhyay, S. and Okumura, M.: Developing Japanese WordNet Affect for Analyzing Emotions. Proceedings of The 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, Portland, Oregon (2011).
11. Santos, C. N. d. and Gatti, M.: Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts. Proceedings of the COLING 2015, Dublin, Ireland (2014).
12. Wang, X., Jiang, W. and Luo, Z.: Combination of Convolutional and Recurrent Neural Network for Sentiment Analysis of Short Texts. Proceedings of COLING 2016, Osaka, Japan (2016).
13. Wang, Y., Huang, M., Zhao, L. and Zhu, X.: Attention-based LSTM for Aspect-level Sentiment Classification. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas (2016).
14. MacCartney, B. and Manning, C. D.: An extended model of natural logic. In Proceedings of the 8th International Conference on Computational Semantics (2009).
15. Benthem, J. v.: Essays in logical semantics. Studies in Linguistics and Philosophy, Springer, 29 (1986).
16. Valencia, V. M. S.: Studies on natural logic and categorial grammar. Ph.D. Thesis, University of Amsterdam (1991).
17. MacCartney, B. and Manning, C. D.: Natural logic for textual inference. In ACL-PASCAL Workshop on Textual Entailment and Paraphrasing (2007).
18. Angeli, G. and Manning, C. D.: NaturalLI: Natural Logic Inference for Common Sense Reasoning. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar (2014).