MCU at NTCIR: Chinese Fact Validation via SVM Context Ranking

Yu-Chieh Wu
Department of Communication and Management
Ming-Chuan University
Taipei, Taiwan
wuyc@mail.mcu.edu.tw

Tzu-Yu Liu
Department of Statistics and Information Science
Fu-Jen University
Taipei, Taiwan
tzuyu2.5@gmail.com

Yue-Shi Lee
Department of Computer Science and Information Engineering
Ming-Chuan University
Taoyuan, Taiwan
leeys@mail.mcu.edu.tw

Jie-Chi Yang
Graduate Institute of Network Learning Technology
National Central University
Taoyuan, Taiwan
yang@cl.ncu.edu.tw

ABSTRACT
Validate factoid description in text is the subtask of finding the textual entailment relation between the given hypothesis and unlabeled raw corpus. By means of integrating multiple natural language processing units, higher performance could be reasonably achieved. In this paper, we propose a context ranking model-based and trainable framework under the condition of part-of-speech tagging information is available. We first revise in-house word segmentation method via auto-deriving thesaurus from Wiki. Then a language-model-based passage retriever is used to find the initial retrieval result. The context ranking model is then extracting features and re-ranks the result. The official results indicate the effectiveness of our method. In terms of accuracy, our method achieves 39.27% for Traditional Chinese FV task (second place).

Categories and Subject Descriptors
D.3.3 [**Programming Languages**]: Language Constructs and Features – *abstract data types, polymorphism, control structures.* This is just an example, please use the correct category and subject descriptors for your submission. The ACM Computing Classification Scheme: http://www.acm.org/class/1998/

General Terms
Your general terms must be any of the following 16 designated terms: Algorithms, Management, Measurement, Documentation, Performance, Design, Economics, Reliability, Experimentation, Security, Human Factors, Standardization, Languages, Theory, Legal Aspects, Verification.

Keywords
Textual entailment, paraphrasing, question answering, passage retrieval

1. INTRODUCTION
Recognizing textual entailment relations receives a great attention in recent years. In this year, NTCIR-RITE creates a new challenge in validating facts given Wiki corpus. The original textual entailment recognition task aims to identify, given two text snippets *t* and *h*, whether *t* entails *h* or not (where *t* means the entailing text and *h* is the hypothesis or the entailed text), while in this new challenge, the goal is to validate whether the Wiki entails the given query. This task is very competitive and raised many text mining techniques, such as Natural Language Processing (NLP) (Manning and Schutze, 1999), Information Extraction (IE), Chinese Text Processing (CTP), Machine Learning (ML) etc. Textual entailment (aka paraphrasing) provides useful information for downstream purposes. Examples include, question answering (Voorhees, 2001; Oh et al., 2007), sentence compression, text summarization, and sentence rephrasing.

NTCIR RITE opens a very early competition on the task of Asian text entailment. It comes up with four different languages, English, Japanese, and (simplified and traditional) Chinese. Participants have to choose FV (fact validation) or SV (system validation) or partial of them and submit the result. SV is the extension of prior RITE tasks by identifying linguistic phenomena. FV is to label the relation of the given fragment. In this year, we only focus on FV task for traditional Chinese.

Chinese textual entailment is a new open research issue. There are fewer literatures about this topic. Huang et al. (2011) presented a complex Chinese textual entailment recognition system. Due to the lack of traditional Chinese syntactic parser, they convert the text into simplified Chinese for parsing. Furthermore, they propose many heuristics to correct the Chinese word segmentation errors and numeric text normalization. They employed the LibSVM (Lin et al., 2005) to learn to find the textual entailment relation. As reported by (Huang et al., 2011), the most useful feature is the "tree mapping" which requires a parser. In English textual entailment (Androutsopoulos and Malakasiotis, 2010), a set of approaches were proposed. For example, the logic proofer (Tatu and Moldovan, 2007), machine learning-based (Li et al., 2007; Malakasiotis, 2009), similarity-based (Malakasiotis and Androutsopoulos, 2007; Wang and Neumann, 2007), syntactic similarity-based (Wan et al., 2006) and hybrid approaches (Tatu and Moldovan, 2007). However, those methods are quite difficult to port to Chinese. The biggest challenge is that there is no explicit word boundary between words. Also, the resources (like parser, thesaurus) for Chinese is limited.

In this paper, we propose a context ranking model based on our prior work in the RITE tasks. Our method works with very limited resources. Only Chinese word segmentation and POS tagging are required. Our method combines both statistical and lexical features. We propose a set of features for learners. Some of features are shallow syntactic pattern-based with only POS tag information while some of them are estimated from the training data. For short, we achieve the second place in the traditional Chinese FV task.

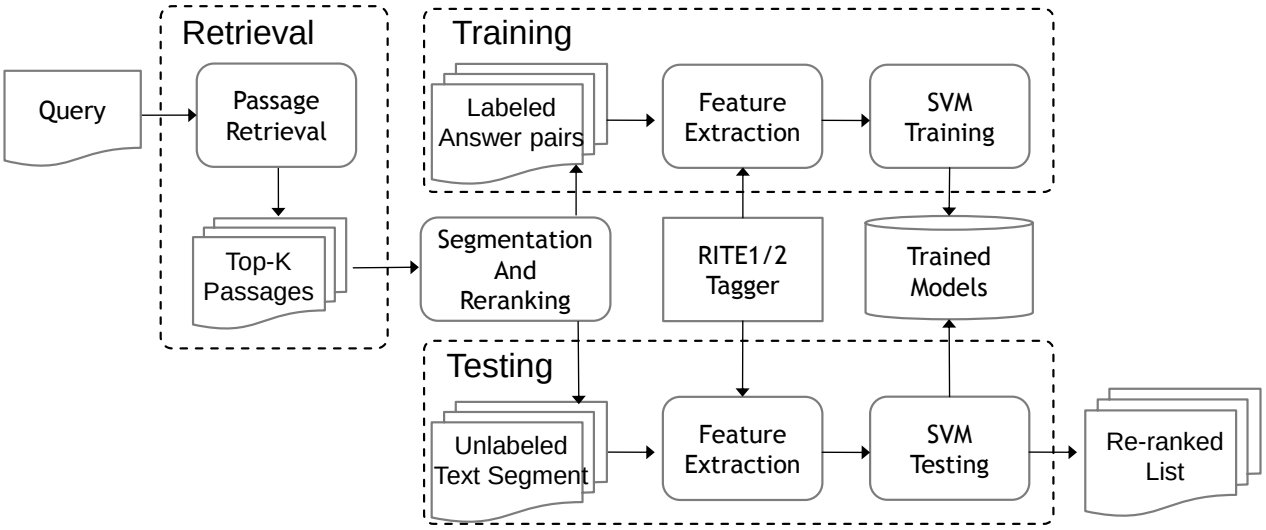

Figure 1: Overall System Flowchart

2. FRAMEWORK

Figure 1 shows the proposed fact validation system used in the NTCIR-RITE FV task this year.

2.1 Retrieval Component

The first component (retrieval) is to query the Wiki corpus and output top-K ranked list for downstream purposes. Directly indexing the entire wiki document is not a good idea, since a topic-based smaller unit, such as passage is more suitable to reduce the searching space. We therefore segment the original wiki document into a series of passages before indexing. To identify the passage boundary, we simply use the pattern, "== sub title ==" to segment the passage which gives the natural paragraph boundary by the users. For each passage, we still append the original document title to the passage in order to keep track of the source.

By following our prior work on the question answer (Wu and Yang, 2008), the retrieval component is a two-pass procedure which consist of initial retriever and the second phase reranker. In this paper, we adopt the Lemur toolkit[1] as the initial passage retriever. Given the input query, we simply tokenize the Chinese sentence into unigram-level unit and input to the lemur. Then the lemur returns top-100 related passages for the second phase reranker. For the retrieval algorithm, we chose the language model with two-stage smoothing. All the parameters were defaultly set for all experiments.

2.2 Segmentation and Reranking

Passage is much larger than sentence in which the number of words is much more than query words. Also, the coverage of a passage is much larger than sentences. To reduce the size of passage, we further segment the passage into a set of text segment which words is limited into 64 Chinese characters and English words. The procedure is listed below.

a. Segment the passage into a set of sentences by the three Chinese characters "。, ？, ！"
b. Start forming a new text segment with first sentence
c. Add next sentence to the text segment
d. Check whether the current text segment size meets the threshold (64 words)
e. If d is false, then append the sentence into current text segment
f. Otherwise, creating another new text segment
g. Repeat (c) to (f) until no new sentence is added

Reranking

After segmenting passages, we further rerank the text segments to reduce the noise. The reranker used in this paper is mainly derived from our prior work (Wu and Yang, 2008) which optimizes the best matching order. It compares the query string with each text segment and generates the best match order by the dynamic programming. The mapping is restricted to one-to-one mapping. The original reranking algorithm requires IDF statistics (inverse document frequency) per match phrase. To avoid from performing Chinese word segmentation over the entire Wiki corpus, we simply weight on certain POS tags, such as Noun and Verb. That is, we first apply the Chinese word segmentation and POS tagging to the given query, and use the POS information (plus position) as the IDF statistics to the reranker. Then, the reranker estimates the matched degree by taking the matched words with associated IDF value into account. Consequently, top-20 text segments were generated and preparing to be validate by the SVM learner in the next stage. In the training phase, the 20 pairs with labels (C, E, or U) is inputting to the SVM, while in the testing, the SVM adopted the trained model and classifies each text segment.

3. RECOGNIZING ENTAILMENT

Our SVM learner is very similar to the original RITE tasks, recognizing textual entailment relations between the query and the retrieved text segment. In this paper, we revise our previous method and implement it to classify whether the text segment has the entailment to the query.

[1] http://www.lemurproject.org/lemur/retrieval.php

2.3 Preprocessing

Text mining in Chinese is quite difficult than most western languages, such as English due to the word information is not available in text. There is no explicit word boundary between words in Chinese text. To resolve this, a Chinese word segmentation tool is needed. It plays an important role the preprocessing step since word information provides the basic concept in term-level for downstream applications. In addition, the POS tag information also gives basic syntactic structures in text. However, there are few Chinese word segmentation tools for our purpose, in this paper, we revise our in-house CMM-based Chinese word segmentation and POS tagging method (Wu et al., 2008, 2010a, 2010b, 2010c).

2.4 Text Normalization

A set of Chinese words share the same meanings as Arabic numbers, such as 伍 equals 5. Also the holomorphy words need to be normalized. However, directly transform these words into numbers is not a good idea, some words might be partial of a person name. To solve this, the normalization process only deals with a small set of POS tags. For Neu (number) and Nd (date) words, Chinese numerical words are directly converted to digits. For example, 1 -> 1, 二->2, 叁-> 3, etc.

There are still some complex Chinese words express numbers, such as 二十一- > 21. A simple rule is designed to solve this. If a specified Chinese word is find (十、廿、卅、百、千、萬), a left-right search is also applied. For all Chinese numeric words that locates on the left hand-side of the specified word, the numeric words were converted using the above text normalization method and multiply the specified word. Similarly, for all the right hand side Chinese numeric words were normalized and plus the left hand side numbers.

2.5 Feature Construction

In this paper, we construct four feature types, statistical, lexical, and thesaurus features. The first two feature types were mainly derived from our prior works (Wu et al., 2011, 2013). The statistical feature is designed to quantize the match/mismatch degree between the two segments. We further add two addition feature, entailment distribution to this type. The entailment distribution is the probability distribution over all possible entailment labels predicted by RITE1/RITE2 taggers. For the second type, we merely adopt the output labels of the two taggers. Below, we list the used features in this paper.

Type I

- Length difference (character-level)
- Length difference (word-level)
- Character match ratio in s_1
- Character match ratio in s_2
- Word match ratio in s_1
- Word match ratio in s_2
- POS match ratio in s_1
- POS match ratio in s_2
- Pattern match ratio in s_2
- Pattern match ratio in s_2
- Reversed pattern match ratio in s_2
- Reversed pattern match ratio in s_2
- Minimum number difference
- Entailment distribution (using RITE 1 tagger)
- Entailment distribution (using RITE 2 tagger)

Type II

- Matched POS tags
- Matched Bi-POS tags
- Mismatched POS tags
- Matched Verb tags
- Mismatched Verb tags
- Mismatched Verb words
- Entailment Label (using RITE 1 tagger)
- Entailment Label (using RITE 2 tagger)

Here, the pattern is predefined as the specified POS bigram and trigrams. We define the following six patterns.

Noun+Verb, Verb+Noun, Noun+Noun, Noun+Verb+Noun,
Verb+Noun+Verb, Noun+Noun+Noun

Even the six patterns are defined to find the matched statistics. We also reverse the *order* for each pattern. That is, the reversed patterns can be used to find the contradiction sentence pairs. To enhance the results, both word and POS tag were used to represent the pattern. For example, the first pattern, Noun+Verb, the word bigram and POS bigram were extracted. In total, there 6*2(POS and Word)*2(plus reverse order) = 24 patterns were extracted.

2.6 Thesaurus Feature

To expand the results in contradiction and entailment types, we further added thesaurus which links the relations between terms. There are three types of the lexicons were adopted, namely, Ciling, positive/negative/negation word set (Wu et al., 2008), and Hownet synonyms. Those lexicons were used as a simple mapping process. Below, we list the used mapping features.

Type III

- The number of matched negation words
- The number of matched positive words
- The number of matched negative words
- The number of matched synonyms
- The number of matched related words
- The number of matched antonyms

2.7 Classification Algorithm

We adopt the SVM (Vapnik, 1995) to learn to classify the testing example. SVM is a kernel-based classifier which can solve non-linear separable problems. Given a set of training examples,

$$(x_1, y_1), (x_2, y_2), ..., (x_n, y_n),\ x_i \in \Re^D,\ y_i \in \{+1, -1\}$$

where x_i is a feature vector in D-dimension space of the i-th example, and y_i is the label of x_i either positive or negative. The training of SVMs is to minimize the following objective function (primal form, soft-margin (Vapnik, 1995):

$$\text{minimize}: W(\alpha) = \frac{1}{2}\vec{W}\cdot\vec{W} + C\sum_{i=1}^{n} Loss(\vec{W}\cdot x_i, y_i) \quad (1)$$

The loss function indicates the loss of training error. Usually, the hinge-loss is used (Keerthi and DeCoste, 2005). The factor C in (1) is a parameter that allows one to trade off training error and margin size. To classify a given testing example X, the decision rule takes the following form:

$$y(X) = sign((\sum_{x_i \in SVs} \alpha_i y_i K(X, x_i)) + b) \quad (2)$$

The α_i is the weight of non-zero weight training example x_i (i.e., α_i>0), and b denotes as a bias threshold of this decision. *SVs* means the support vectors and obviously has the non-zero weights

of α_i. $K(X, x_i) = \phi(X) \cdot \phi(x_i)$ is a pre-defined kernel function that might transform the original feature space from $\Re^D$ to $\Re^{D'}$ (usually D<<D').

2.8 Training/Testing

Training the SVM-based ranker is mainly done by categorizing training examples into three categories, "entailment", "contradiction", and "unknown". To label the training data, the passage retriever ranks top 25 text fragments for annotators. Then, the human annotators start labeling each instance and assign the labels to each text fragment. The training data is mainly derived from three sources: (1) NTCIR-11 official provided training set, (2) RITE1/2 training/testing pairs (using "F", "B", and "C"), and (3) and (3) 100 contradiction sentences from the online resources[2]. Our human labeler randomly select the 100 false descriptions to support the training examples.

To form the entailment and contradiction training examples, the human labeler requires to read the retrieved text one by one and finally determine whether the text is "entailing" or "conflicting" the query. For example, if the rank6 text is labeled as "entailment", then the query and the text is assigned to the "entailment" category, while the others are assigned to "unknown" category. The case is also true to "contradiction" type. By summing up all, there are ~30275 training examples to SVM. However, the SVM is a binary classifier, to port to multiclass, we employ the so-called one-versus-all method. We use the classifier ensemble algorithm to fuse multiple SVM classifiers (Wu, 2013).

In testing, the passage retriever also output top-25 ranked list. We use the same feature extraction method and classified by the SVM for each text segment. If "entailment" or "contradiction" is found by the SVM ranker, then it stops searching and labels "entailment" to the query. If there is no any text segment in the top-25 ranked list belonging to the two categories, then the unknown label is used to tag this query.

4. EXPERIMENTAL RESULTS

4.1 Settings

The RITE1/2 taggers were mainly derived from our previous work on the NTCIR-RITE 1 and 2 (Wu et al., 2011, 2013). For the classification algorithm, in this paper we adopt the LibSVM (Chang and Lin, 2011) and SVMlight (Joachims, 1998) for training and testing. LibSVM and SVMlight have different strategies for solving multiclass problem. The default setting of LIBSVM is one-versus-one multiclass SVM, while we implement our one-versus-all strategy for SVMlight.

The kernels used in this paper are: 1) polynomial kernel with degree 2 and 2) RBF kernel with Gaussian is verified by 0.1~0.03. Our word segmentation and POS tagger is also derived from our prior works (Wu, 2014b; Wu et al., 2008). To enhance the segmentation consistency, we add the local AV information and the Wiki title dictionary to correct our initial tagger (Wu, 2014a, 2014b).

[2] http://yamol.tw/main.php and http://content.edu.tw/senior/history/ks_rs/test.htm/

4.2 Results

At the beginning, we report our RITE 1/2 taggers. Table 1 and Table **2** list the official result on the RITE 1 and RITE 2 data sets. The first row indicates the best official result to the dataset. In RITE 1 MC, our method achieves better accuracy than the official best result (Wu et al., 2011). In RITE 2 MC task, our method showed even better accuracy than the best approach.

Table 1: System performance of our RITE1 tagger on the NTCIR-RITE 1 MC task

Method	Testing Data
Best reported result	53.60%
RBF kernel	51.44%
Polynomial kernel	53.44%
Ensemble method	53.77%

Table 2: System performance of our RITE2 tagger on the NTCIR-RITE 2 MC task

Method	Testing Data
Best reported result	56.64%
RBF kernel	58.79%
Polynomial kernel	58.45%
Ensemble method	58.91%

Table 3 lists the official results on the traditional Chinese FV task in the RITE-2014. Table 4 shows the compared results of the FV task on the same dataset.

Table 3: Official results on the traditional Chinese FV task

Type	F1	Precision	Recall
Entailment	37.33	71.79	25.23
Contradiction	30.33	86.05	18.41
Unknown	50.15	34.76	90
Acc.	43.07		
MacroF1	39.27		

As shown in Table 3, it is observed that our method showed high recall rate in "unknown" type while high precision rate in the "contradiction" type. However, our method performed poorly in the recall rates. This reveals that the final SVM ranker gives high confidence in "contradiction" and "unknown" on a small fragment of the data and fails to recognize the entire set. This is mainly caused by the lack of training data. It is trusted that by feeding with more training examples, the recall rates could be enhanced. Second, our method showed very competitive result as well as the best approach (39.27 v.s. 39.51 in MacroF1). Even though the used lexical features (with POS tag) is far away from the other systems that adopted grammar features such as dependency structures and semantic roles, the performance is very satisfactory. We left the integration of deep syntax and semantic features as future work.

Table 4: Comparison to the other competitors

Team	MacroF1	Acc.	Improvment Rate (MacroF1)	Improvment Rate(Acc.)
I*-02	39.51	44.70	-0.61	-3.65
I*-05	39.36	44.54	-0.23	-3.30
This Paper	39.27	43.07	0.00	0.00
W*-02	38.08	41.92	3.13	2.74
I*-01	38.04	42.90	3.23	0.40
I*-03	37.72	42.41	4.11	1.56
I*-04	37.69	44.05	4.19	-2.22
W*-01	35.94	39.97	9.27	7.76
K*-01	33.97	36.38	15.60	18.39
G*-02	31.07	35.89	26.39	20.01
G*-03	30.88	36.70	27.17	17.36
G*-01	29.07	34.91	35.09	23.37
G*-04	28.73	35.89	36.69	20.01
G*-05	26.02	36.87	50.92	16.82

5. CONCLUSION

Recognizing Inference in Text is an important and new research topic in recent years. Fewer research papers addressed on the Chinese language. This paper presents a context ranking model based on machine learning framework for RITE task this year. Using only Chinese word segmentation and POS tagging information, this method achieves the second place in official competition result. As summary, it achieves 39.27% Micro F1 in Chinese FV task.

In the future, we plan to explore deep Wiki knowledge to represent the text. Also, if the grammar parser is available, we will adopt the parse features.

ACKNOWLEDGMENTS

The authors acknowledge support under MOST Grants 103-2221-E-130-004-

REFERENCES

[1] I. Androutsopoulos and P. Malakasiotis. 2010. A survey of paraphrasing and textual entailment methods. Journal of Artificial Intelligence Research, 38: 135-187.

[2] C. C. Chang and C. J. Lin. 2011. LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2(27):1-27.

[3] D. Giampiccolo, B. Magnini, I. Dagan and B. Dolan. 2007. The third PASCAL recognition textual entailment challenge. In ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pp. 1-9.

[4] W. C. Huang, S. H. Wu, L. P. Chen, and C. K. 2011. Chinese textual entailment analysis. In Proceedings of the 23rd Conference on Computational Linguistics and Speech Processing.

[5] T. Joachims. 1998. Text categorization with support vector machines: learning with many relevant features. In Proceedings of the European Conference on Machine Learning, pages 137-142.

[6] S. Keerthi and D. DeCoste. 2005. A modified finite Newton method for fast solution of large scale linear SVMs. Journal of Machine Learning Research. 6: 341-361.

[7] B. Li, J. Irwin, E. V. Garcia, and A. Ram. 2007. Machine learning based semantic inference: experiments and observations at RTE-3. In ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pp. 159-164.

[8] P. Malakasiotis. 2009. Paraphrase recognition using machine learning to combine similarity measures. In Proceedings of the ACL-IJCNLP 2009 Student Research Workshop, pp. 27–35.

[9] P. Malakasiotis and I. Androutsopoulos. 2007. Learning textual entailment using SVMs and string similarity measures. In Proceedings of ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pp. 42-47.

[10] C. D. Manning and H. Schutze 1999. Fundations of statistical natural language processing. The MIT Press, London.

[11] H. J. Oh, S. H. Myaeng, and M. G. Jang. 2007. Semantic passage segmentation based on sentence topics for question answering. Information Sciences, 177(18): 3696-3717.

[12] M. Tatu and D. Moldovan. 2007. COGEX at RTE3. In ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pp. 22-27.

[13] V. N. Vapnik. 1995. The Nature of Statistical Learning Theory. Springer.

[14] E. M. Voorhees. 2001. Overview of the TREC 2001 question answering track. In Proceedings of the 10th Text Retrieval Conference, 42-52.

[15] S. Wan, M. Dras, R. Dale, and C. Paris. 2006. Using dependency-based features to take the “parafarce” out of paraphrase. In Proceedings of the Australasian Language Technology Workshop, pp. 131-138.

[16] R. Wang and G. Neumann. 2007. Recognizing textual entailment using sentence similarity based on dependency tree skeletons. In ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pp. 36-41.

[17] Y. C. Wu. 2014a. A Top-down Information Theoretic Word Clustering Algorithm for Arbitrary Phrase Recognition. Information Sciences, 275:213-225.

[18] Y. C. Wu. 2014b. A Sparse L2-Regularized Support Vector Machines for Efficient Natural Language Learning. Knowledge and Information Systems, 39(2): 305-328.

[19] Y. C. Wu. 2013. Integrating statistical and lexical information for recognizing textual entailments in text. Knowledge-Based Systems. 40: 27-35.

[20] Y. C. Wu and J. C. Yang. 2008. A Robust Passage Retrieval Algorithm for Video Question Answering.

IEEE Transactions on Circuits and Systems for Video Technology, 18(10): 1411-1421.

[21] Y. C. Wu, Y. S. Lee, and J. C. Yang. 2008. Robust and efficient multiclass SVM models for phrase pattern recognition. Pattern Recognition, 41(9): 2874-2889.

[22] Y. C. Wu, J. C. Yang, and Y. S. Lee. 2013. Learning to Recognize Chinese Textual Inference via Shallow POS patterns and Classifier Ensemble. Proceedings of the 10th NTCIR workshop meeting on evaluation of information access technologies, pp. 567-572.

[23] Y. C. Wu, C. J. Lee, and Y. C. Chen. 2011. MCU at NTCIR: A Resources Limited Chinese Textual Entailment Recognition System. In Proceedings of the 9th NTCIR workshop meeting on evaluation of information access technologies, pp.567-572.

[24] Y. C. Wu, L. W. Yang, J. Y. Shen, L. Y. Chen, and S. T. Wu. 2008. Tornado in Multilingual Opinion Analysis: A Transductive Learning Approach for Chinese Sentimental Polarity Recognition. Proceedings of the 7th NTCIR workshop meeting on evaluation of information access technologies, pp. 301-306.