Soft Comput (2017) 21:311–330
DOI 10.1007/s00500-015-1629-1

FOCUS

Exploring lexical, syntactic, and semantic features for Chinese textual entailment in NTCIR RITE evaluation tasks

Wei-Jie Huang · Chao-Lin Liu

Published online: 26 March 2015

Abstract We computed linguistic information at the lexical, syntactic, and semantic levels for Recognizing Inference in Text (RITE) tasks for both traditional and simplified Chinese in NTCIR-9 and NTCIR-10. Techniques for syntactic parsing, named-entity recognition, and near synonym recognition were employed, and features like counts of common words, statement lengths, negation words, and antonyms were considered to judge the entailment relationships of two statements, while we explored both heuristics-based functions and machine-learning approaches. The reported systems showed their robustness by simultaneously achieving second positions in the binary-classification subtasks for both simplified and traditional Chinese in NTCIR-10 RITE-2. We conducted more experiments with the test data of NTCIR-9 RITE, with good results. We also extended our work to search for better configurations of our classifiers and investigated contributions of individual features. This extended work showed interesting results and should encourage further discussions.

Keywords Textual entailment recognition · Negation and antonyms · Near synonym recognition · Named-entity recognition · Dependency parsing · Trained heuristic functions · Support-vector machines · Linearly weighted models · Decision trees

Communicated by C.-S. Lee.

W.-J. Huang · C.-L. Liu
Department of Computer Science,
National Chengchi University, Taipei, Taiwan
e-mail: s951553@gmail.com

C.-L. Liu (✉)
Graduate Institute of Linguistics,
National Chengchi University, Taipei, Taiwan
e-mail: chaolinliu@gmail.com; chaolin@nccu.edu.tw

1 Introduction

Recognizing textual entailment[1] (RTE) (Dagan et al. 2006) has become a major research topic in natural language processing (NLP) in the past decade (Watanabe et al. 2013a). Given a pair of statements, *text* (T) and *hypothesis* (H), the most basic format of an RTE task is to determine whether H is true when T is true; namely, whether or not T entails H. A more challenging format is to determine whether T and H are contradictory statements (Dagan et al. 2009). More recently in PASCAL RTE-6,[2] NTCIR-10 RITE-2,[3] and NTCIR-11 RITE-VAL,[4] researchers investigated and evaluated methods for identifying statements in a collection, e.g., a corpus like Wikipedia, which are relevant to a given statement T, where relevancy includes both entailment, paraphrase, and contradiction.

The RTE tasks are relevant and applicable to many NLP applications, including knowledge management (Tsujii 2012). If a statement entails another in a collection of statements, then one may not need to consider both statements to produce a concise summary of the collection, so recognizing entailments is useful for automatic text summarization (Lloret et al. 2008; Tatar et al. 2009). Similar reasons apply to how recognizing entailment can be applied to question answering systems (de Salvo et al. 2005). When a question entails another, the recorded answer to the previous question may be useful for answering the new question. RTE can also be useful for judging the correctness of students' descriptive answers in assessment tasks. It is rare for stu-

[1] The pronunciations and translations of all Chinese strings mentioned in this paper are provided in the Appendix.

[2] http://www.nist.gov/tac/2010/RTE/.

[3] http://www.cl.ecei.tohoku.ac.jp/rite2.

[4] https://sites.google.com/site/ntcir11riteval/.

dents to respond to questions with statements that are exactly the same as the instructors' standard answers. It is also not practical to expect instructors to list all possible ways which students may answer a question. In such cases, recognizing paraphrase relationships between students' and instructors' answers becomes instrumental (Nielsen et al. 2009). We have also applied RTE techniques to enable computers to take reading comprehension tests that are designed for middle school students (Huang et al. 2013).

Dagan et al. (2009) provided an overview of the approaches for RTE. Treating RTE as a classification task is an obvious option, where different systems consider various factors to make the final decisions. Due to the availability of the training data in RTE activities, machine learning-based approaches are common. Researchers design methods to utilize different levels of linguistic, including syntactic and semantic, information provided in the given statement pairs to judge their relationships. Transformation-based methods offer interesting alternatives for the RTE tasks. If a statement can be transformed into another via either syntactic rewriting (Bar-Haim et al. 2008; Stern et al. 2011; Shibata et al. 2013) or logical inference procedures (Chambers et al. 2007; Takesue and Ninomiya 2013; Wang et al. 2013; Watanabe et al. 2013b), then the statements may be highly related. In addition to using the information conveyed by the given statements, external information like common sense knowledge and ontology about problem domains can strengthen the basis on which entailment decisions are made (de Salvo et al. 2005; Stern et al. 2010).

The first corresponding event of PASCAL RTE for Japanese and Chinese took place in NTCIR-9, and was named RITE as the acronym for "Recognizing Inference in Text" (Shima et al. 2012). NTCIR-10 continued to host RITE-2 for Japanese and Chinese, and had, respectively, ten and nine teams participating in the traditional and simplified Chinese subtasks (Watamabe et al. 2013). All of these participants considered different combinations of linguistic information as features to determine the entailment relationships of statement pairs. Most of them employed support vector machines as the classifiers.

There were different subtasks in NTCIR-9 RITE and NTCIR-10 RITE-2. The binary classification (BC) subtask required participants to judge whether or not T entails H. In this paper, we will focus only on the BC subtasks in the NTCIR RITE tasks, as we believe that the BC subtask is the most fundamental subtask of them all.

In NTCIR-10 RITE-2, the best performing team in the BC subtask for traditional Chinese (CT) adopted a voting mechanism (Shih et al. 2013). The best performing team in the BC subtask for simplified Chinese (CS) employed an alignment-based strategy (Wang et al. 2013). We (Huang and Liu 2013) trained heuristic functions to achieve second best performance in the BC subtasks for both CT and CS. The best team outperformed us in the BC subtask for CT by only 0.7 % in the F1 measure. Chang et al. (2013) embraced decision trees as the classifier but did not achieve an impressive performance.

For obvious reasons, all participating systems in NTCIR-10 RITE-2 used some forms of linguistic features to make decisions. As may be expected, different systems considered different sets of features and applied them in different ways. We computed lexical, syntactic, and semantics information about the statement pairs to judge their entailment relationships. The linguistic features were computed with public tools and machine-readable dictionaries, including the Extended HowNet[5] (Chen et al. 2010). Preprocessing steps for the statements included conversion between simplified and traditional Chinese, Chinese segmentation, and converting formats of Chinese numbers. We employed such linguistic information as (1) words that were shared by both statements; (2) synonyms, antonyms, and negation words; (3) information about the named entities of the statement pairs; and (4) similarity between parse trees and dependency structures, etc.

The performance of our approaches was sufficiently robust that we achieved the second best scores in both CT and CS subtasks. Since each participating team could submit running results of three different configurations, we actually experimented with our models that we built by training heuristic functions and support vector machines (SVMs). Our best results were achieved by the trained heuristic functions, achieving second position in the BC subtasks for both CT and CS. Our SVM-based models achieved the third best score in the BC subtask for CT, but dropped to 12th position in BC subtask for CS.

We have extended our work after participation in NTCIR-10 RITE-2. We ran grid searches of larger scales to find the best combinations of parameters and features for the classification models. In general, conducting the grid searches helped us build better models. However, the experimental results also provide interesting and seemingly perplexing material for further discussion in the paper. We also tested our systems with the test data for the BC subtasks of NTCIR-9 RITE, and found that we were able to achieve better performance than the best performer in NTCIR-9 RITE tasks.

We explain the preprocessing of the text material and extraction of their linguistic features in Sect. 2, examine the constructions of the heuristics-based and machine learning-based classifiers in Sect. 3, present and discuss the experimental results in Sect. 4, review and deliberate on some additional observations in Sect. 5, and wrap up this paper in Sect. 6.

[5] http://ehownet.iis.sinica.edu.tw/.

2 Major system components

In this section, we describe components of our running systems, including the preprocessing steps and the extraction of fundamental linguistic features.

2.1 Preprocessing

In this subsection, we explain the preprocessing steps: traditional-to-simplified Chinese conversion, numeric format conversion, and Chinese segmentation.

2.1.1 Traditional-to-simplified Chinese conversion

We relied on Stanford NLP tools[6] to do Chinese segmentation and named-entity recognition. As those tools were designed to perform better for simplified Chinese, we had to convert traditional Chinese into simplified Chinese. We converted words between their traditional and simplified forms of Chinese with an automatic procedure which relied on a tool in Microsoft Word. We did not design or invent a conversion dictionary of our own, and the quality of conversion depended solely on Microsoft Word.

There are two major methods for converting between traditional and simplified Chinese text. The simpler option is just to do character-to-character conversion, e.g., changing "電腦軟體的品質很重要"[7] to "电脑软体的品质很重要". A more sophisticated and better conversion is to do word-to-word conversion, changing this sample statement to "计算机软件的质量很重要". This latter conversion includes the simplified Chinese words, i.e., "计算机", "软件", and "质量" that are used in the training of the Stanford tools, so is more likely to lead to better system performance. Microsoft Word offers the second type of conversion as much as it can, and we understand that Microsoft Word might not convert all traditional Chinese words perfectly to their simplified counterparts, e.g., the result of converting "工業技術水準" is "工业技术水准". "工业技术**水平**" is a preferred conversion. However, Microsoft Word is a good and accessible current choice.

2.1.2 Numeric format conversion

There are multiple ways for people to write numbers in English text, e.g., sixteen vs. 16. In Chinese, there are at least three ways to write numbers in text, e.g., "3", "三", and "參" for the number "3". There are also specific characters to express specific numbers, e.g., "廿" and "卅" for 20 and 30, respectively. In addition, there are simplified ways to express relatively small numbers, e.g., "三十二" for 32 but "十二" for 12. In the latter case, "一十二" is more formal but is rarely used.

To streamline our handling of numbers in Chinese statements, we employed regular expressions to capture specific strings and convert them to Arabic numerals. The conversions need special care for some extraordinary instances. For instance, one may not want to convert "朝九晚五" to "朝 9 晚 5" or convert " 舉一反三" to " 舉 1 反 3".

2.1.3 Chinese string segmentation

We employed the Stanford Word Segmenter[8] (Chang et al. 2008) to segment Chinese character strings into word tokens. Unlike most alphabetical languages in which words are separated by spaces, Chinese text strings do not have delimiters between words. In fact, Chinese text did not use punctuation marks until modern times. In the field of natural language processing, converting a Chinese string into a sequence of Chinese words is called segmentation (or tokenization) of Chinese.

A major challenge of Chinese segmentation is that different segmentations of a given Chinese string can represent very different meanings of the original string. We can segment the string "研究生命還有多少年" in two different ways: {"研究生命", "還有", "多少年"} or {"研究生", "命", "還有", "多少年" }. Adopting the former segmentation, the translation of the original Chinese string is "how many more years can one do research". Adopting the latter will lead to "how many more years can the graduate student survive". To most native speakers of Chinese, the former segmentation is much more natural, but the latter is not unacceptable. In the 2012 Bakeoff for Chinese segmentation, the best performing system reached an F1 measure slightly shy of 95 % (Duan et al. 2012).

2.2 Lexical semantics

2.2.1 Lexical resources and computation for Chinese synonyms

The number of words shared by statement pairs is the most commonly used feature to judge entailment. Identifying words that are shared literally is a direct way to compute word overlaps. Indeed, in previous RTE and RITE events, organizers provided baseline systems which calculated character overlaps to determine entailment (Bar-Haim et al. 2006; Stern et al. 2011).

In practice, people may express the same or very similar ideas with synonyms and near synonyms, so their identification is also very important. The following statements are

[6] http://nlp.stanford.edu/software/index.shtml.

[7] The pronunciations and translations of all Chinese strings mentioned in this paper are provided in the Appendix.

[8] http://nlp.stanford.edu/software/segmenter.shtml.

very close in meaning though they do not use exactly the same words.

(1) Tamara is **reluctant** to **raise** this question.
(2) Tamara **hesitates** to **ask** this question.

Translating this pair into Chinese will also show the importance of identifying synonyms.

(3) Tamara 對於**提出**這一**問題**感到**猶豫**
(4) Tamara 對於**詢問**這一**問題**顯得**遲疑**

The literature has seen abundant ways to compute synonyms for English, particularly those that computed the similarity between words based on WordNet[9] (Budanitsky and Hirst 2006). In contrast, we have yet to find a good way to compute synonyms for Chinese.

To compute synonyms for a given word, we rely on both existing lexicons and computing methods. We acquired a dictionary for synonyms and antonyms[10] from the Ministry of Education (MOE) of Taiwan. This MOE dictionary lists 16,005 synonyms and 8625 antonyms.

We could employ the extended HowNet[11] (E-HowNet), which can be considered as an extended WordNet for Mandarin Chinese, to look up synonyms of Chinese words. E-HowNet contains 88,079 traditional Chinese words in the 2012 version, and can provide synonyms of Chinese words, so we could use the list of synonyms directly. We will find 38 synonymous words[12] which carry the concept of "hesitate" in E-HowNet. In this particular case, we would be able to tell that "遲疑" in statement (3) and "提出" in statement (4) are synonymous with the list in E-HowNet. However, "詢問" in statement (3) does not belong to the synonym list[13] of "猶豫" in statement (4). "提出" is similar to "raise" in English. One can raise a question or a concern, so "raise" alone does not necessarily relate to asking questions.

We could also use the definitions for words in E-HowNet to estimate the relatedness between two Chinese words by their taxonomical relations and semantic relations (Chuang et al. 2012; Chen 2013; Huang and Liu 2013). In this work, we converted the definition of a word into a "definition tree", e.g., Fig. 1, according to the taxonomy in E-HowNet. Each node

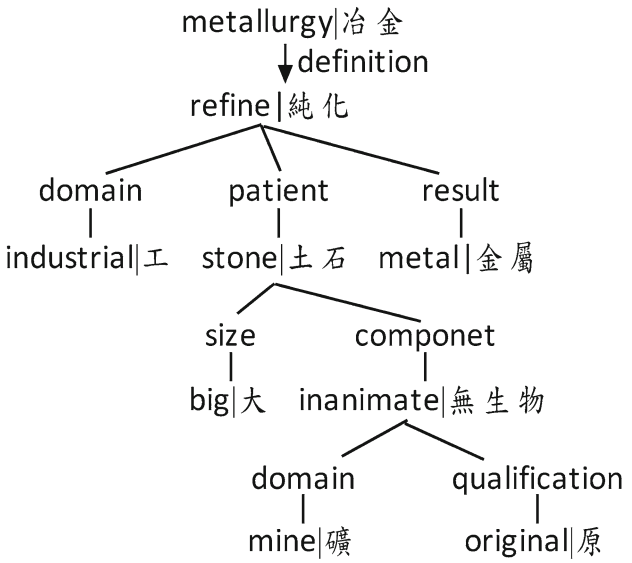

Fig. 1 A definition tree for "冶 金" (metallurgy)

represents a primitive unit, a function word, or a semantic role. Considering each internal node in a definition tree as a root, we built a collection of subtrees of the definition tree. In Fig. 1, there are 15 nodes.

The DICE coefficient[14] between the collections of subtrees of two definition trees is used to measure the degree of relatedness of two definitions. Given two collections, e.g., X and Y, the DICE coefficient is defined in Eq. (1), where $|X|$ is the number of elements in X.

$$\mathrm{DICE}(X,\ Y) = \frac{2\,|X \cap Y|}{|X| + |Y|} \tag{1}$$

Due to the definition, a DICE coefficient must fall in the range of [0, 1]. Two definitions will be considered anonymous if their DICE coefficient is larger than a threshold, for which we chose to use 0.88 based on a small-scale experiment.

Computing Chinese synonyms only with information in dictionaries is an imperfect method. Chinese text contains out-of-vocabulary (OOV) words a lot more frequently than English text. For these OOV words, dictionary-based methods cannot always help.

2.2.2 Chinese antonyms and negation words

We consider two ways to express opposite meanings. The first is *antonyms*, e.g., "good" vs. "bad"; and the second is through *negation* words, e.g., "good" and "not good".

We relied on the lists of antonyms provided by the MOE dictionary (cf. Sect. 2.2.1). Since there are only 8625 words

[9] http://wordnet.princeton.edu/.

[10] http://dict.revised.moe.edu.tw/.

[11] http://ehownet.iis.sinica.edu.tw/.

[12] 三心二意, 心猿意馬, 彷徨, 投鼠忌器, 沈吟, 沉吟, 委決不下, 狐疑不決, 首鼠兩端, 動搖, 徘徊不前, 逡巡, 游移, 猶猶豫豫, 猶疑, 猶疑不決, **猶豫**, 猶豫不決, 搖擺不定, 當斷不斷, 滯足不前, 裹足, 裹足不 前, 踟躕, 踟, 踟躕, 踟躕不前, 踟躕不進, **遲疑**, 遲疑不決, 舉棋不定, 舉棋未定, 瞻前顧後, 躊躇, 顧忌, 躕, 觀望不前, 搖擺.

[13] 叩問, 打探, 打聽, 扣問, 咨, 咨詢, 查詢, 討教, 追問, 問, 問到, 問津, 問訊, 問起, 問話, 問道, 探問, 探詢, 尋問, 提問, 敢問, 發問, 詰問, 詢, **詢問**, 詢答, 徵詢, 請旨, 請教, 質詢, 諮諮, 詢諮, 諏訊.

[14] http://en.wikipedia.org/wiki/S%C3%B8rensen%E2%80%93Dice_coefficient.

in the antonym lists in the dictionary, we can handle only a very small number of antonyms at this moment.

We created a list of negation words based on our own judgment. This list of negation words include "無", "未", "不", "非", and "沒有". Note that we consider "無", "未", "不", and "非" to be negation words only when they are individual words after segmentation. Hence, we will handle words like "並非" correctly. This list allows us to find that statements (5) and (6) (used in NTCIR-10 RITE-2) have opposite meanings.

(5) 千禧年危機俗稱Ｙ２Ｋ危機或千禧蟲
(6) 千禧年危機並非Ｙ２Ｋ危機或千禧蟲

We could also handle other negation words like "無法", "未能", "不行", and "不能". However, this heuristic list is as yet unable to handle all possible Chinese negation words correctly. A more complex word like, "無可厚非", would need special attention in our system. A direct application of our heuristic list will treat this word as two negations, but this word is not really related to negation.

2.2.3 Named entity and verb recognition

Among parts of speech in almost all languages, nouns and verbs are the essential parts for understanding the core meanings of sentences. Information about named entities such as persons, locations, organizations, and time are crucial for inferring relationships between statements. A software tool for named entity recognition (NER) not only annotates words in a sentence as nouns but also subcategorizes them as persons, locations, organization names and time specifications. Although current technologies for NER do not offer perfect performance, being able to carry out NER even partially paves a way to handle typical questions regarding the five Ws (What, When, Where, Why, Who). We employed S-MSRSeg, which is a tool for named entities recognition developed by Microsoft Research (Gao et al. 2005).

Verbs provide information about the actions or states that a given sentence describes. Recognizing verbs for a sentence pair is thus useful. We employed the Stanford parser (Levy and Manning 2003) to do the tagging of parts of speech. Although it is possible to consider sub-categorization of verbs, we did not do so in the current study.

2.3 Syntactic features

We parsed the Chinese statements with the Stanford parser (Levy and Manning 2003) to obtain the parse trees and the part-of-speech (POS) tags for words. A parse tree of a sentence reveals important information about the meaning of the sentence. At this moment, we used the parsing results to do two types of comparisons. The first was to compare the similarity between the parse trees of T and H with the same method (the DICE coefficient) that we used to compare the definition trees of different senses as explained in Sect. 2.1.1. We also compared the collections of POS tags of two sentences, particularly the tags for verbs.

Based on our experience, the Stanford parser works better for simplified Chinese than for traditional Chinese. Hence, we converted statements of traditional Chinese into simplified Chinese before the parsing step in our procedures (cf. Sect. 2.1.1).

We noticed that the Stanford parser did not always produce the best or even correct parse trees for the given statements. The parser ranked candidate parse trees with probabilistic models, and produced the trees with leading scores. Although we could request more than one parse tree for a given statement, we chose to use only the top-ranked tree for computational efficiency of our systems.

2.4 Semantic features

It is preferable to employ higher level information about statement pairs to judge their entailment relationships. After considering information available at the lexical and syntactic levels, semantic features immediately came to mind. However, there are multiple ways to define and represent sentential semantics. Frame semantics is a conceivable choice (Fillmore 1976; Burchardt et al. 2009), for instance. In this work, we explored an application of dependency structures (Chang et al. 2009).

Linguists consider the context of words a very important factor to define meaning. "You shall know a word by the company it keeps" (Firth 1957) or similar arguments (e.g., Firth 1935; Harris 1954) are commonly cited in courses on linguistics. "One sense per discourse, one sense per collocation" (Yarowsky 1995) appears in the literature in computational linguistics very frequently. For this reason, using vector space models to capture contextual information has become one of the standard approaches in both natural language processing and information retrieval.

In our work, we explored an application of dependency structures to capturing the contextual information in a sentence. There are different ways to apply the dependency structures for inferring entailment relationships, and we note that Day et al. also employed the tree-edit distances of dependency structures in NTCIR-10 RITE-2 (Day et al. 2013).

We illustrate our methods with a short English example, "We consider dependency structures for inferring textual entailment", to make the example more easily understandable to non-Chinese speakers. We list the typed and collapsed dependencies of this statement below. A dependency relation is expressed in the format of *relation-name* (*governor, dependent*), where both governor and dependent are words appended with their positions in the sentence.

Table 1 Matrix form for encoding dependency structures

	We	Consider	Dependency	Structures	Inferring	Textual	Entailment
We	0	1	0	0	0	0	0
Consider	0	0	0	0	0	0	0
Dependency	0	0	0	1	0	0	0
Structures	0	1	0	0	0	0	0
Inferring	0	1	0	0	0	0	0
Textual	0	0	0	0	0	0	1
Entailment	0	0	0	0	1	0	0

nsubj(consider-2, We-1)
root(ROOT-0, consider-2)
amod(structures-4, dependency-3)
dobj(consider-2, structures-4)
prepc_for(consider-2, inferring-6)
amod(entailment-8, textual-7)
dobj(inferring-6, entailment-8)

We can ignore the root node and build a matrix to encode the direct relationships between words, as shown in Table 1. The column headings show the governors, and the row headings show the dependents. A cell will be 1 if there is a relationship from the dependent to the governor. Hence, ignoring the relation name, the cell (We, consider) is 1 because of nsub(consider-2, We-1). Notice that the matrix is not symmetric because of the functions of words in different relationships.

The matrix, denoted by R, encodes the holistic relationships between words in a statement, and can be considered a way to represent the context of words in a given statement. There are many similar applications of such matrices in computer science, e.g., for modeling connectivity between web pages (Page et al. 1998) and for modeling traffic networks (Liu and Pai 2006).

As R encodes only the direct relationships between words, we can compute the powers of R to explore the indirect relationships between the words. For example, a "1" in the second power of R, R^2, shows that there is a one-step indirect relationship between two words. If we compute the second power of the matrix in Table 1, we will find that the cell with "dependency" as the row heading and with "consider" in the column heading is 1—suggesting the idea of "consider dependency" in the statement. When we compute higher powers of R, we will find fewer "1"s in the matrices because there are fewer word pairs with very remote indirect relationships.

Based on such observations, we explored the possibility of encoding the sentential context with the union of the powers of R for a statement. In the reported experiments in this paper, we chose to compute the XR matrix, defined in Eq. (2), for a given statement. A cell in XR will be 1 if the cell at the corresponding positions in any of the first five powers of R is 1.

$$XR = R \cup R^2 \cup R^3 \cup R^4 \cup R^5 \tag{2}$$

3 Classification methods

Although machine learning-based algorithms are the most conceivable method for classification problems including the recognition of textual entailment (Dagan et al. 2009), the size of training data available at NTCIR-10 RITE-2 was not large enough to make us feel comfortable to just take this intuitive avenue. Hence, in addition to applying support vector machines, we also tried to come up with our own parameterized heuristic functions to make classification decisions. The parameters would be tuned with the training data, so, technically, we can still consider our first approach as a machine learning-based method.

3.1 Trained heuristic functions

We explain the individual factors that we considered in our heuristic function in the following subsections.

3.1.1 Word overlap

Character overlap was used in the baseline systems in previous RTE (Bar-Haim et al. 2006) and RITE evaluations (Stern et al. 2011). Perhaps, for this reason, word overlap may be the most common feature used by participating teams in these events.

Since our goal is to judge whether T entails H, we would like to know the portion of words in H that also appear in T. In addition, we consider word overlap rather than character overlap. The difference is important because Chinese words are consisted of Chinese characters. Some words may contain just one character, but most others contain multiple characters. Hence, we must segment the given statements to compute their word overlap. The word overlap between T and H is defined in Eq. (3), where $W(T)$ and $W(H)$, respectively, denote the bags of words of T and H after the segmentation step (cf. Sect. 2.1.3). We borrow the symbol for set intersection, $\cap$, to indicate the common part of two bags of words. We represent the size of a bag of words by surrounding the notation for the bag with vertical bars, e.g., $|W(T)|$.

$$\mathrm{WOL}(T, H) = \frac{|W(T) \cap W(H)|}{|W(H)|} \tag{3}$$

Assume that we segment the statements in sentences (5) and (6) to obtain (7) and (8), respectively. Their word overlap will be 6/7, and their character overlap will be 14/16.

(7) 千禧年 危機 俗稱 Y2K 危機 或 千禧蟲
(8) 千禧年 危機 並非 Y2K 危機 或 千禧蟲

3.1.2 Missing named entities

The intuition is: if some named entities in H are missing in T, then it may be less likely for T to entail H. Hence, we measured the missing named entities (MNE) in (4), where count(T, H) is the number of named entities that appear in H but not in T. Namely, let NE(T) and NE(H), respectively, denote the collections of named entities in T and H. count(T,H) is then defined as NE(H)\(NE(T) ∩ NE(H)).

$$\mathrm{MNE}(T, H) = \alpha \times \mathrm{count}(T, H) \tag{4}$$

The value of α would be selected with a training procedure.

3.1.3 Imbalanced negations

The following statement pair appeared in the development set of NTCIR-10 RITE-2.

(9) 若望保祿二世是四百五十多年來第一位非義大利籍的教宗
(10) 若望保祿二世是四百五十多年來第一位義大利籍的教宗

These statements convey opposite meanings because of the negation word "非". Hence, we consider a penalty term for imbalanced negations, IN(T,H) in Eq. (5), based on the number of negation words (cf. Sect. 2.2.2) for both T and H, where |NEG(T)| and |NEG(H)| are the number of negation words in $T$and H, respectively.

$$\mathrm{IN}(T, H) = \begin{cases} \beta, & |\mathrm{NEG}(T)| \neq |\mathrm{NEG}(H)| \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The value of β would be selected with a training procedure.

3.1.4 Occurrence of antonyms

As an extension of the consideration of negation words, the occurrence of antonyms, in T, of some words, in H, indicate that the statement pairs are unlikely to entail one another. Hence, we considered the following factor in our heuristic function.

$$\mathrm{OA}(T, H) = \begin{cases} \gamma, & \{t \text{ is an antonym of } h \\ & \quad | \ t \in W(T), h \in W(H)\} \neq \emptyset \\ 1, & \text{otherwise} \end{cases} \tag{6}$$

The value of γ was in the range of [1, 2] and would be selected with a training procedure.

3.1.5 An integrated heuristic decision function

Putting Eqs. (3), (4), (5), and (6) together, we have the following score function for whether T entails H.

$$s(T, H) = \frac{\mathrm{WOL}(T, H) - \mathrm{MNE}(T, H) - \mathrm{IN}(T, H)}{\mathrm{OA}(T, H)} \tag{7}$$

Relying on intuitive hunches, we subtract or divide the scores for negative factors, and we admit that the arrangements were not scientific and not normative.

In some cases, the order of named entities influences the entailment relationships. The following statement pair shows an extreme example.

(11) 台灣出口至印度成長 28.6 %
(12) 印度出口至台灣成長 28.6 %

The word overlap of them is perfect, but they express almost opposite information. For such observations, we also considered the order of named entities (ONE) in our heuristics when $s(T,H)$ is large enough.

We define a penalty term for the order of named entities in (8).

$$\mathrm{ONE}(T, H) = \begin{cases} \delta^{\tau}, & s(T, H) \geq \lambda \\ 1, & \text{otherwise} \end{cases} \tag{8}$$

The value of τ is the number of pairs of named entities in T and H that have different orders. In sentences (11) and (12), the named entities "台灣" and "印度" have different orders, so τ will be one in this instance. The values of δ, which is in the range of [1, 2], and λ would be selected with the training data.

Integrating Eqs. (7) and (8), we obtain the following heuristic decision function that we used in the NTCIR-10 RITE-2 task. If score(T, H) exceeds a chosen threshold, E, we will determine that T entails H.

$$\mathrm{Score}(T, H) = \frac{s(T, H)}{\mathrm{ONE}(T, H)} \tag{9}$$

3.1.6 A brief critical review

In the previous subsections, we introduced individual terms for the final version of the heuristic decision function in Eq. (9). We tried to justify the influences of each individual term by isolated observations, so each individual term may look reasonable. Unfortunately, real-world statements can be complex, and may demand deeper understanding of the statement pairs to determine whether or not their entailment relationships hold.

Table 2 List of candidate features for machine learning-based classifiers

ID	Type	Definition	ID	Type	Definition
F1	Num	$\frac{\lvert W(T)\cap W(H)\rvert}{\lvert W(H)\rvert}$	F2	Num	$\lvert \{h \lvert h \in \text{NE}(H), h \notin \text{NE}(T)\} \rvert$
F3	Num	τ defined in Sect. 3.1.5	F4	Num	$\lvert \{t \in \text{ANT}(h) \lvert t \in W(T), h \in W(H)\} \rvert$
F5	Bool	1, if $\lvert\text{NEG}(T)\rvert = \lvert\text{NEG}(H)\rvert$; 0, else	F6	Num	$\lvert W(T)\rvert$ and $\lvert W(H)\rvert$
F7	Bool	1, if $\lvert W(T)\rvert > \lvert W(H)\rvert$; 0, else	F8	Num	DICE coefficient between the subtrees of the parse trees of T and H
F9	Num	$\frac{\lvert\text{NE}(T)\cap\text{NE}(H)\rvert}{\lvert\text{NE}(H)\rvert}$	F10	Num	$\lvert\text{NE}(T)\rvert$ and $\lvert\text{NE}(H)\rvert$
F11	Num	$\lvert\text{NEG}(T)\rvert$ and $\lvert\text{NEG}(H)\rvert$	F12	Num	$\frac{\lvert\text{NEG}(T)\cap\text{NEG}(H)\rvert}{\lvert\text{NEG}(H)\rvert}$
F13	Num	$\lvert\{t \in \text{SYN}(h) \lvert t \in W(T), h \in W(H)\}\rvert$	F14	Num	$\frac{\lvert\{t\in\text{SYN}(h) \lvert t\in W(T), h\in W(H)\}\rvert}{\lvert W(H)\rvert}$
F15	Num	$\frac{\lvert\text{VERB}(T)\cap\text{VERB}(H)\rvert}{\lvert\text{VERB}(H)\rvert}$	F16	Num	$\frac{\lvert\text{XR}(T)\cap\text{XR}(H)\rvert}{\lvert\text{XR}(H)\rvert}$
F17	Num	$\lvert\text{VERB}(T)\rvert$ and $\lvert\text{VERB}(H)\rvert$			

Consider the following statement pair which appeared in the NTCIR-10 RITE-2 development set.

(13) 台灣出口至印度成長 28.6 %
(14) 印度從台灣進口成長率可達 28.6 %

We have a pair of antonyms, i.e., "出口" and "進口". We also observe that a pair of named entities have reversed order in the statement pair, i.e., "台灣" and "印度". The existence of antonyms and the reversed order of a named entities pair are considered negative factors against the holding of entailment relationships in the previous subsections, when we discussed them separately. However, in this case, when both negative factors occur, they cancel each other out, and this statement pair can be considered as a pair of paraphrased statements. As a consequence, our heuristic function would fail to work for them.

Despite such practical challenges, Eq. (9) is indeed the decision function that we employed to achieve the second positions in the BC subtasks for both TC and SC in NTCIR-10 RITE-2. We will provide details about its performance shortly.

3.1.7 Machine learning methods

We considered more features when we ran experiments that employed techniques of support vector machines, decision trees, and linearly weighted models.

3.1.8 The candidate features

We considered 17 candidate features that are listed in Table 2, where we use X to denote a sentence and x to denote a word in X in the following definitions.

1. "Num" and "Bool", respectively, denote "numeric" and "Boolean" in the Type column.
2. $W(X)$: the collection of words of a sentence X (after segmentation).
3. $S \cap T$: the collection of elements that appear in both collection S and collection T.
4. $|S|$: the number of elements in the collection S.
5. $\text{NE}(X)$: the collection of named entities in a sentence X.
6. $\text{ANT}(x)$: the collection of antonyms of a word x.
7. $\text{NEG}(X)$: the collection of negation words in a sentence X.
8. $\text{SYN}(x)$: the collection of synonyms or near synonyms of x.
9. $\text{VERB}(X)$: the collection of POS tags of the verbs in X (cf. Sect. 2.3).
10. $\text{XR}(X)$: the *XR* matrix of X (cf. Sect. 2.4).

Many of the features listed in Table 2 are derivations of those basic features that we discussed in Sects. 2.2, 2.3, and 2.4. Others were selected due to similar rationalities, so we do not repeat the same reasoning, and explain their derivations only briefly below.

F1: This is the word overlap discussed in Sect. 3.1.1.
F2: This is the count(T, H) in Sect. 3.1.2.
F3: This feature is defined in Sect. 3.1.5.
F4: This feature is similar to the word overlap that we discussed in Sect. 3.1.1, except that we consider the antonyms here.
F5: This feature measures whether T and H have the same number of negation words (cf. Sect. 3.1.3).
F6: We consider the number of words in T and H. These are typical features for all RITE and RTE systems (Shima et al. 2012).
F7: We examine whether T is longer than H. This is also a typical feature for RITE and RTE systems (Shima et al. 2012).
F8: This is Eq. (1) (cf. Sect. 2.2.1).
F9: This feature calculates the overlap of named entities that we discussed in Sect. 2.

F10: These features record the quantities of named entities in T and H (cf. Sect. 2).
F11: These features record the quantities of negation words in T and H (cf. Sect. 2.2.2).
F12: This feature calculates the overlap of negation words in T and H (cf. Sect. 2.2.2).
F13: This feature records the overlap of synonyms in T and H (cf. Sect. 2.2.1).
F14: This feature records the proportion of synonyms in H (cf. Sect. 2.2.1).
F15: Mimicking the principle of calculating word overlaps, this feature records the proportion of common verbs in T and H (cf. Sect. 2.2.3).
F16: This feature was discussed in Sect. 2.4.
F17: Analogous to the principle of computing word counts in T and H (F6), these features record the number of verbs in T and H.

Notice that we would consider counts for both T and H when we adopted F6, F10, F11, and F17. These counts are numeric features for the statements, and we thought it would be unreasonable to consider just the count for an individual statement in the statement pairs.

3.1.9 The classifiers: SVMs, decision trees, and linearly weighted models

We employed the libSVM library for SVMs (Chang and Lin 2011) and Weka for decision trees and linearly weighted functions for classification (Witten et al. 2011).

We used the radial basis function as the kernel function in libSVM, and tuned the parameters with standard methods recommended by Chang and Lin (2011). The values of the features were also normalized as recommended.

We utilized the packages for learning decision trees and linearly weighted models with the default settings in Weka, and did not attempt to change the parameters of the packages.

When using the linearly weighted functions to judge the entailment relationship of a statement pair, we computed the score of the statement pair with a linearly weighted function. This function considered the features that are listed in Table 2. A statement pair whose score was larger than 0.5 was considered to have an entailment relationship. We let the learning package find the coefficients that would optimize the classification results. In essence, this procedure of using linearly weighted functions is quite similar to our using heuristic functions in Sect. 3.1.

4 Empirical evaluations

We applied the aforementioned features and classification methods to participate in the BC subtask of the NTCIR-10 RITE-2 task, and achieved the second positions for both traditional Chinese (TC) and simplified Chinese (SC). Since the winning teams of the TC and SC tracks were different, we have good reason to believe that our system is relatively more robust in its performance.

In this long section, we provide information about the data sources in Sect. 4.1, and explain the methods for typical RITE evaluations in Sect. 4.2. The results of our participation in NTCIR-10 are reported in Sect. 4.3. Due to time constraints, we did not choose the parameters for our heuristic functions (cf. Sect. 3.1) systematically when we participated in the evaluation tasks. We have extended our work afterwards, and the results are presented in Sects. 4.4 through 4.7. The purpose of conducting these new experiments was to check how different approaches and different data sets influenced the observed results. Some additional discussions about the results are provided in Sect. 5.

4.1 Data sources

By participating in the NTCIR-10 RITE-2 task, we obtained a development data set for training purposes and a test data set for formal runs. We could also download the test data set for NTCIR-9 RITE. Table 3 shows the statistics of the provided data for RITE tasks.

The development data set contains pairs of statements that are annotated with the correct answers as to whether or not the first statement entails the second. We list a positive pair (with a "Y" label) and a negative pair (with an "N" label) below. In Table 3, we show the number of "Y" pairs and "N" pairs.

```
<pair id="2" label="Y">
  <t1>香港的主權和領土是在 1997 由英國歸還給中國的。</t1>
  <t2>1997 年香港回歸中國</t2>
</pair>
<pair id="3" label="N">
  <t1>一九九一年波斯灣戰爭結束時，雅辛又帶著家人移居約旦。</t1>
  <t2>波斯灣戰爭發生於 1991 年</t2>
</pair>
```

4.2 Evaluation metrics

We use the evaluation metrics adopted by the NTCIR-10 RITE-2 tasks. They are standard definitions of accuracy, precision rate, recall rate, and the F1 measure (Watanabe et al. 2013a).

Accuracy is the proportion of the correct classifications among all predicted classifications. *Y-precision* is the proportion of true Y pairs among all pairs that are classified as Y. *Y-recall* is the proportion of true Y pairs among all pairs that are actually Y. *N-precision* is the proportion of true N pairs among all pairs that are classified as N. *N-recall* is the proportion of true N pairs among all pairs that are actually N.

Table 3 Quantities of statement pairs in the RITE and RITE-2 data sets.

Category	NTCIR-9 RITE test		NTCIR-10 RITE-2 development		NTCIR-10 RITE-2 test	
Language	TC	SC	TC	SC	TC	SC
Y pairs	450	263	716	528	479	422
N pairs	450	144	605	286	402	359
Total	900	407	1321	814	881	781

Table 4 Parameters for the heuristic decision functions

	E	α	β	γ	λ	δ	Accuracy %
TC	0.57	0.28	0.24	2.0	0.85	2.0	72.90
SC	0.56	0.25	0.24	2.0	0.85	2.0	71.25

The *F1 measure* is defined as the division of the product of 2, precision, and recall over the sum of precision and recall. *MacroF1* is the average of the F1 measures of the Y category and the N category.

4.3 NTCIR-10 RITE-2 evaluation task

Table 4 shows the parameters that we used for our heuristic functions to participate in NTCIR-10 RITE-2. The meanings of E, α, β, γ, λ, and δ were explained in Sect. 3.1. We chose the values of these parameters based on observed results of some experiments that we conducted with the development set. A statement pair, T and H, whose score(T,H), defined in Eq. (9), exceeded the value of E would be considered to have the entailment relationship. The aim at our training stage was to optimize accuracy.

At the time when we submitted our results, we wanted to study the effects of considering synonyms in computing word overlap. Hence, we submitted two runs of classifications that were obtained by two procedures that differed only in whether synonyms were considered as overlapped words. The formal run that was obtained when we considered synonyms was *MIG-2*, and the formal run that intentionally ignored synonyms was *MIG-1*.

When we had to submit the results for formal runs, we had just begun to try machine learning-based models. At that moment, we only tried SVMs and decision trees with a specific set of features. We employed F1, F2, F6, F7, F8, F10, F11, F12, F13, and F14 for TC (cf. Table 2), and F1, F2, F6, F7, F8, F9, and F10 for SC. Using the 10-fold cross-validation on the development set with SVM models, we observed 71.46 % in accuracy for TC and 75.55 % for SC, and we submitted a run with these configurations. The results obtained with such SVM models were coded *MIG-3*.

Table 5 lists the results of MIG-1, MIG-2, and MIG-3, along with the results of the best performing team, IASL-2 (Shih et al. 2013), for TC. Table 6 lists the results of MIG-1, MIG-2, and MIG-3, along with the results of the best performing team, bcNLP-3 (Wang et al. 2013), for SC. We do not show percentage signs in Tables 5 and 6 and all the remaining tables to save space.

The performance values of IASL-2 and MIG-2 are really close to each other in Table 5. In contrast, although MIG-2 achieved the second best performance for SC, there were big gaps between the performance values of bcNLP-3 and MIG-2 in Table 6.

The MacroF1 values in Tables 5 and 6 indicate that considering synonyms in calculating word overlap helped MIG-2 to perform better than MIG-1 in the evaluation of both TC and SC.

Many may be disappointed that using SVM-based models did not achieve the best performance. None of the leading teams, including IASL-2, bcNLP-3, and MIG-2, used SVM. The best performing systems that used SVMs are MIG-3 in Table 5 for TC and CYUT-3 (Wu et al. 2013) in Table 6 for SC. Both achieved third place. IMTKU-1 (Day et al. 2013) used SVM-based models as well, and performed similarly with MIG-3 in TC subtasks. We suspect that the relatively small size of the available data for training, listed in Table 3, may have contributed to this phenomenon. We will discuss this issue further in Sect. 5.2.

4.4 More experiments for the heuristic functions

We relied on limited experimental results to select the combinations of the parameters for the heuristic function, and chose to use the combination listed in Table 4. After NTCIR-10, we had the opportunity to run a more exhaustive grid search for the parameters.

Using the settings in Table 4 as seeds, we chose a range for each of the parameters, and ran experiments on all possible combinations of the parameters with the development set of NTCIR-10 RITE-2. The ranges and increments for all parameters are listed in Table 7. Notice that the ranges contain the values which we listed in Table 4. Although the selections of the ranges and increments remained arbitrary, the searched region was quite large, and we had to conduct more than 317 million experiments to search the region for both TC and for SC. Hence we ran the experiments more than 634 million times with the development set.

Table 5 Partial results of BC subtask for TC in NTCIR-10 RITE-2

Rank	Team-ID	MacroF1	Acc.	Y-F1	Y-Prec.	Y-Rec.	N-F1	N-Prec.	N-Rec.
1	IASL-2	67.14	67.76	71.66	68.64	74.95	62.63	66.48	59.20
2	MIG-2	67.07	67.54	70.99	69.03	73.07	63.14	65.51	60.95
3	MIG-3	66.99	67.54	71.23	68.74	73.90	62.76	65.85	59.95
4	IMTKU-1	65.99	66.29	69.16	68.80	69.52	62.83	63.22	62.44
6	MIG-1	65.42	65.61	67.94	68.88	67.01	62.91	61.93	63.93

Table 6 Partial results of BC subtask for SC in NTCIR-10 RITE-2

Rank	Team-ID	MacroF1	Acc.	Y-F1	Y-Prec.	Y-Rec.	N-F1	N-Prec.	N-Rec.
1	bcNLP-3	73.84	74.65	78.43	72.58	85.31	69.25	78.25	62.12
2	MIG-2	68.09	68.50	71.72	69.64	73.93	64.45	66.97	62.12
3	CYUT-3	67.86	68.12	70.74	70.16	71.33	64.98	65.63	64.35
6	MIG-1	65.71	65.81	67.56	69.33	65.88	63.87	62.11	65.74
12	MIG-3	57.19	63.64	73.80	60.42	94.79	40.59	81.51	27.02

Table 7 Ranges and increments for the grid search

Parameter	Range	Increment
E	[0.40, 0.70]	0.01
α	[0.05, 0.35]	0.01
β	[0.05, 0.35]	0.01
γ	[1.00, 2.00]	0.10
λ	[0.55, 0.90]	0.05
δ	[1.00, 2.00]	0.10

Table 8 Best combinations of parameters for TC

Configuration ID	E	α	β	γ	λ	δ	Accuracy
C1	0.54	0.1	0.27	1.8	0.85	1.9	73.05
C2	0.56	0.08	0.25	1.0	0.85	1.8	73.13
C3	0.56	0.08	0.25	1.7	0.85	1.8	73.20
C4	0.56	0.09	0.25	1.0	0.85	1.8	73.28
C5	0.56	0.09	0.25	1.7	0.85	1.8	73.35
C6	0.57	0.28	0.24	2.0	0.85	2.0	72.90

Table 9 Best combinations of parameters for SC

Configuration ID	E	α	β	γ	λ	δ	Accuracy
C7	0.40	0.17	0.10	1.1	0.85	1.0	74.69
C8	0.40	0.18	0.07	1.1	0.85	1.0	74.82
C9	0.40	0.18	0.09	1.1	0.85	1.0	74.94
C10	0.40	0.20	0.07	1.1	0.85	1.0	75.06
C11	0.41	0.20	0.06	1.0	0.85	1.0	75.18
C12	0.56	0.25	0.24	2.0	0.85	2.0	71.25

Within the region described in Table 7, we found some combinations of these parameters that would help us achieve higher accuracies than those listed in Table 4. Table 8 lists such new settings for TC, and Table 9 lists such new settings for SC. If we had used an exhaustive search for the parameters, we would have used the combinations in Tables 8 and 9 to participate in NTCIR-10 RITE-2, rather than using the combinations listed in Table 4. Note that we intentionally repeated the settings listed in Table 4 in Table 8, i.e., C6, and in Table 9, i.e., C12, to facilitate comparison between results.

We used the settings in Table 8 to run experiments on the TC test set of NTCIR-10 RITE-2. Recall that when we submitted our classification results for formal runs, MIG-1 did not consider synonyms for counting word overlap, but MIG-2 did. We did not consider synonyms in experiments to obtain the results in Table 10, and considered synonyms to obtain Table 11.

Comparing the MacroF1 values in Table 10 with that of MIG-1 in Table 5, we find that using any of the five new settings would help us achieve better MacroF1 scores, but only marginally. Comparing the MacroF1 values in Table 11 with that of MIG-2 in Table 5, we see that using three of the five new settings would help us improve the MacroF1 scores. Using two of these new settings, i.e., C2 and C4, would actually help us achieve the best MacroF1 in formal runs. Nevertheless, we note that the improvements were not very significant.

We used the settings in Table 9 to run experiments on the SC test set of NTCIR-10 RITE-2. Tables 12 and 13, respectively, list the results of not considering and considering synonyms. Although the new settings achieved better accuracies for the development data than the settings listed in Table 4, they would not provide better performances for the test data.

Considering synonyms in computing word overlaps would lead to better performance for TC subtasks in NTCIR-10

Table 10 Using settings in Table 8 but no synonyms for TC in NTCIR-10 RITE-2

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C1	65.79	66.29	67.91	72.03	64.08	59.45
C2	65.73	66.29	67.70	72.65	64.31	58.71
C3	65.55	65.95	68.01	70.56	63.28	60.45
C4	65.75	66.29	67.77	72.44	64.23	58.96
C5	65.56	65.95	68.08	70.35	63.21	60.70
C6 (MIG-1)	65.42	65.61	68.88	67.01	61.93	63.93

Table 11 Using settings in Table 8 and synonyms for TC in NTCIR-10 RITE-2

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C1	66.79	67.76	67.63	78.08	67.99	55.47
C2	67.46	68.56	67.91	79.96	69.72	54.98
C3	67.12	67.99	68.07	77.45	67.86	56.72
C4	67.25	68.33	67.79	79.54	69.28	54.98
C5	66.92	67.76	67.96	77.04	67.46	56.72
C6 (MIG-2)	67.07	67.54	69.03	73.07	65.51	60.95

Table 12 Using settings in Table 9 but no synonyms for SC in NTCIR-10 RITE-2

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C7	62.05	66.33	62.80	92.42	80.00	35.65
C8	61.98	66.33	62.76	92.65	80.38	35.38
C9	62.24	66.45	62.90	92.42	80.12	35.93
C10	62.67	66.71	63.15	92.18	80.00	36.77
C11	63.94	67.35	63.94	90.76	78.57	39.83
C12 (MIG-1)	65.71	65.81	69.33	65.88	62.11	65.74

Table 13 Using settings in Table 9 and synonyms for SC in NTCIR-10 RITE-2

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C7	59.35	64.92	61.42	94.31	81.95	30.36
C8	59.05	64.66	61.27	94.08	81.20	30.08
C9	59.44	64.92	61.46	94.08	81.48	30.64
C10	60.10	65.30	61.78	93.84	81.43	31.75
C11	61.68	66.33	62.60	93.60	82.00	34.26
C12 (MIG-2)	68.09	68.50	69.64	73.93	66.97	62.12

RITE-2. The corresponding MacroF1 values in Table 11 are better than those in Table 10. In contrast, considering synonyms did not lead to consistent improvements in MacroF1 scores for SC subtasks in NTCIR-10 RITE-2. The corresponding MacroF1 scores in Table 13 are not necessarily higher than those in Table 12.

4.5 More experiments for the machine learning-based models

In Sect. 4.3, we reported results of using an SVM model with a set of features that were chosen based on some small-scale experiments. Since the size of the training data is not large and we have listed only 17 candidate features in Table 2, it is not infeasible for us to use all possible combinations of the 17 features with a classification model to pinpoint the combination that produces the best classification results for the training data. The number of experiments is 2^{17}, which is 131,072.

We actually executed just such a brute-force search for SVMs, decision trees, and linearly weighted functions with the TC and SC development set of NTCIR-10 RITE-2. Tables 14 and 15 list the selected sets of features along with the accuracies observed in the 10-fold cross-validation learning processes, where SVM, DT, and LM, respectively, denote SVMs, decision trees, and linearly weighted models. Recall that, in Sect. 4.3, the selected feature set for SVMs led to 71.46 % in accuracy for TC and 75.55 % for SC at training time, both of which are not very different from their counterparts in Tables 14 and 15.

Comparing Tables 14 and 15, we can see that the best combination of features varies with the language and the nature of the classifiers.

Having identified the best features for different classifiers with the development dataset, we ran the classifiers, which were based on linear models, on the test dataset of NTCIR-10 RITE-2. Tables 16 and 17 list the results for TC and SC, respectively. A comparison between the MacroF1 scores in Table 16 and the MacroF1 of MIG-3 in Table 5 shows that none of the classifiers that used the new feature sets outperformed the SVM model which we used in the TC subtask in the NTCIR-10 RITE-2. On the contrary, the MacroF1 scores in Table 17 are significantly better than the MacroF1 of the MIG-3 in Table 6. Nevertheless, even after such improvements, these new results would not be good enough to be listed among the top 5 results for the SC subtask in NTCIR-10 RITE-2.

4.6 Evaluations with NTCIR-9 RITE test data

We reused the classification models that we trained with the NTCIR-10 RITE-2 development dataset to predict the entailment of the test data for NTCIR-9 RITE. According to Shima et al. (2012), the best accuracy scores achieved by software for the TC and SC were 66.11 and 77.64 %, respectively.

We used our heuristic functions with the settings listed in Table 8 to predict the entailment relationships of the TC test dataset of NTCIR-9 RITE. Again, we ran two sets of experiments, differing in whether or not synonyms were used

Table 14 Feature selection with the TC development set of NTCIR-10 RITE-2

Model ID	Feature ID	Accuracy
SVM-1	F1, F2, F3, F4, F5, F6, F8, F9, F12, F14	71.99
SVM-2	F1, F2, F4, F5, F6, F8, F9, F11, F12, F14	71.84
DT-3	F1, F2, F3, F5, F7, F8, F12, F13, F15	71.78
DT-4	F1, F2, F3, F5, F7, F8, F10, F13, F15	71.74
LM-5	F1, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12, F13, F14, F15, F16, F17	72.98
LM-6	F1, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12, F13, F15, F16, F17	72.75

Table 15 Feature selection with the SC development set of NTCIR-10 RITE-2

Model ID	Feature ID	Accuracy
SVM-7	F1, F2, F5, F6, F7, F9, F13, F16	75.80
SVM-8	F1, F2, F3, F5, F6, F7, F16	75.80
DT-9	F1, F2, F5, F7, F11, F12	76.44
DT-10	F1, F2, F4, F5, F7, F8, F11, F12	76.40
LM-11	F1, F2, F3, F4, F5, F7, F8, F9, F10, F12, F13, F15, F16, F17	77.40
LM-12	F1, F3, F4, F5, F7, F8, F9, F10, F11, F12, F13, F14, F15, F16, F17	77.27

Table 16 Results of using LM and the best feature sets for TC test set of NTCIR-10 RITE-2

Model ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
LM-5	64.60	64.81	67.94	66.81	61.22	62.44
LM-6	64.10	64.36	67.30	67.01	60.89	61.19

Table 17 Results of using LM and the best feature sets for SC test set of NTCIR-10 RITE-2

Model ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
LM-11	62.16	65.94	62.87	90.28	76.57	37.33
LM-12	62.05	65.81	62.81	90.05	76.14	37.33

Table 18 Using settings in Table 8 but no synonyms for TC in NTCIR-9 RITE

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C1	73.18	73.44	69.57	83.33	79.22	63.56
C2	73.29	73.56	69.63	83.56	79.44	63.56
C3	73.59	73.78	70.34	82.22	78.61	65.33
C4	73.52	73.78	69.89	83.56	79.56	64.00
C5	73.82	74.00	70.61	82.22	78.72	65.78
C6	73.83	73.89	71.81	78.67	76.41	69.11

in computing word overlap, and the results are listed in Tables 18 and 19.

The data in Tables 18 and 19 show that our accuracy scores were better than the best score achieved by the systems which participated in the TC evaluation task of NTCIR-9. However,

Table 19 Using settings in Table 8 and synonyms for TC in NTCIR-9 RITE

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C1	71.53	72.11	67.18	86.44	81.00	57.78
C2	71.94	72.56	67.41	87.33	82.02	57.78
C3	72.41	72.89	68.13	86.00	81.02	59.78
C4	72.07	72.67	67.53	87.33	82.08	58.00
C5	72.54	73.00	68.25	86.00	81.08	60.00
C6	73.00	73.22	69.68	82.22	78.32	64.22

Table 20 Using settings in Table 8 but no synonyms for SC in NTCIR-9 RITE

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C7	67.69	75.18	73.82	95.44	82.09	38.19
C8	67.69	75.18	73.82	95.44	82.09	38.19
C9	67.69	75.18	73.82	95.44	82.09	38.19
C10	68.11	75.43	74.04	95.44	82.35	38.89
C11	68.11	75.43	74.04	95.44	82.35	38.89
C12	67.66	71.01	76.36	79.85	59.85	54.86

we also observed that considering synonyms in TC experiments for NTCIR-9 actually decreased the performance of our systems.

We also used our heuristic functions with the settings listed in Table 9 to predict the entailment relationships of the SC test dataset of NTCIR-9 RITE. Analogously, we ran two sets of experiments, differing in whether or not synonyms were used in computing word overlap, and the results are listed in Tables 20 and 21.

Table 21 Using settings in Table 8 and synonyms for SC in NTCIR-9 RITE

ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
C7	65.96	74.94	72.80	97.72	88.89	72.80
C8	65.96	74.94	72.80	97.72	88.89	72.80
C9	65.96	74.94	72.80	97.72	88.89	72.80
C10	66.41	75.18	73.01	97.72	89.09	73.01
C11	66.41	75.18	73.01	97.72	89.09	73.01
C12	67.57	71.74	75.52	83.27	62.39	75.52

Table 22 Results of using the best feature sets for TC test set of NTCIR-9 RITE

Model ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
LM-5	71.50	71.89	67.75	83.56	78.55	60.22
LM-6	72.48	72.89	68.39	85.11	80.29	60.67

Table 23 Results of using the best feature sets for SC test set of NTCIR-9 RITE

Model ID	MacroF1	Acc.	Y-Prec.	Y.-Rec.	N-Prec.	N-Rec.
LM-11	71.58	77.64	75.90	95.82	85.33	44.44
LM-12	72.36	78.13	76.36	95.82	85.71	45.83

The statistics in Tables 20 and 21 show that our accuracy scores were not as good as the best score achieved by the systems which participated in the SC subtask of NTCIR-9 RITE. Similar to what we observed in Tables 18 and 19, considering synonyms in SC experiments for NTCIR-9 RITE brought down the performance of our systems.

We used the linear model-based classifier with the best feature sets (cf. Tables 14, 15 in Sect. 4.5) to predict the entailment relationships for the test dataset of NTCIR-9 RITE. Tables 22 and 23 show the results for TC and SC, respectively. Once again, the accuracies for TC were better than the best performing team which actually participated in NTCIR-9 RITE. Moreover, the accuracy achieved by LM-12 was also slightly better than the best accuracy for SC in NTCIR-9 RITE.

4.7 Effects of syntactic and semantic information

In order to study the effects of considering parse trees (F8 in Table 2) and the dependency structures (F16 in Table 2), we intentionally removed F8 and F16 from LM-5 and LM-6 in Table 14 and LM-11 and LM-12 in Table 15. We used LM-5A, LM-6A, LM-11A, and LM-12A to denote these new settings. Table 24 lists the MacroF1 and accuracy scores when we used LM-5A, LM-6A, LM-11A, and LM-12A with linearly weighted models to predict entailment.

Although we hoped that considering higher level linguistic information could make a significant contribution to the scores, the data does not support our hypothesis decisively. Most of the time, considering F8 and F16 made the classification results only relatively and marginally better for simplified Chinese. The effects of considering F8 and F16 were quite arbitrary for test data of traditional Chinese, as indicated by the left side of Table 24.

5 Additional discussions

In this section, we discuss some issues that involve observations obtained in multiple experiments. More specifically, we discuss the implication that was suggested by the experiments reported in Sect. 4. Although one might expect that some approaches should have achieved better performance than others, such expectations might not be realized in the current study. We investigate the issues and elaborate on possible reasons for the gap between the actual results and expected outcomes in this section.

5.1 Y-precision, Y-recall, N-precision, and N-recall

Although we have focused mostly on the effects of using different methods and features on the achieved MacroF1 and accuracy scores, the values of the Y-precision, Y-recall, N-precision, and N-recall are informative for the design of algorithms.

It should be noted that, when handling the statement pairs of simplified Chinese, our methods had high values in Y-

Table 24 Effects of considering syntactic and semantic information indecisive

	Traditional Chinese					Simplified Chinese			
	RITE.Test		RITE-2.Test			RITE.Test		RITE-2.Test	
	MacroF1	Acc.	MacroF1	Acc.		MacroF1	Acc.	MacroF1	Acc.
LM-5	71.50	71.89	64.60	64.81	LM-11	71.58	77.64	62.16	65.94
LM-5A	71.70	72.00	64.51	64.81	LM-11A	70.95	77.15	62.31	65.81
LM-6	72.48	72.89	64.10	64.36	LM-12	72.36	78.13	62.05	65.81
LM-6A	71.32	71.67	64.34	64.70	LM-12A	69.25	75.68	62.03	65.56

Table 25 Performance statistics of teams which participated in both SC and TC subtasks in NTCIR-10 RITE-2

Teams	Simplified Chinese			Traditional Chinese		
	Ranks	MacroF1	Acc.	Ranks	MacroF1	Acc.
JUNLP	24	48.49	48.66	16	48.72	48.81
IASL	10,18	60.45	63.25	1,14	67.14	67.76
MIG	2,6,12	68.09	68.50	2,3,6	67.07	67.54
CYUT	3,7,9	67.86	68.12	12,13,15	55.16	55.16
Yuntech	14,15,16	53.52	59.54	8,9,10	62.31	62.54
IMTKU	13,17,23	54.28	62.74	4,7,17	65.99	66.29
WHUTE	8,11	61.65	66.58	5	65.55	66.29

recall and N-precision and low values in N-recall in Sects. 4.4 and 4.5. After using the training methods, our methods showed a tendency to grant entailed relationships to statement pairs. We suspect that this phenomenon may have resulted from the imbalanced portions of Y-pairs and N-pairs in the development set (cf. Table 3).

5.2 Performance of SVM-based systems

Indeed, it is not surprising that the quality of training data influenced the performances of the trained models. The amount of data available for training may have also affected the performances of teams which adopted supported vector machines (SVMs) as their classifiers. Table 25 shows some statistics of the performance of all of the teams which participated in the BC subtask for both simplified and traditional Chinese in NTCIR-10 RITE-2. Since each team could submit up to three runs of their systems, a team would have as many results as the runs they submitted. The "MacroF1" and "Acc." columns show the highest MacroF1 and accuracy achieved by the teams.

Among the seven teams, only IASL (Shih et al. 2013) did not use SVMs, and MIG (Huang and Liu 2013) used SVMs in one of their three runs. The other five teams used SVMs as their classifiers, and only CYUT (Wu et al. 2013) achieved better performance in simplified Chinese than in traditional Chinese. Although MIG's best performance in simplified Chinese is better than its best performance in traditional Chinese, as shown in Table 25, MIG's performance in simplified Chinese is actually poorer than its performance in traditional Chinese when MIG used an SVM-based classifier (cf. MIG-3 in Tables 5, 6).

5.3 Effects of specific features on experiments with real test data

Comparing the experimental results discussed in Sects. 4.3, 4.4, and 4.5, we found that, overall, using systematic ways to search for parameters and features offered us more chances to achieve better performance than relying on results of intuitively selected experiments to build an inference system.

We have also attempted to compare many experimental results that were influenced by whether or not we considered synonyms in computing word overlap in Sect. 4. The following statement pair of NTCIR-10 RITE-2 provides an example of the need to consider synonyms. One needs to recognize the synonymous relationship between "聽力" and "聽覺" to correctly handle this pair.

(15) 噪聲對動物也有很大的影響，降低動物聽力
(16) 噪聲對動物的聽覺有很大的影響

Nevertheless, experimental results showed that considering synonyms only helped improve our performance in the TC experiments in NTCIR-10 RITE-2. Similar results were not observed in other experiments that we reported in Sects. 4.4 and 4.6. This may have resulted because the test data did not include many instances that really needed synonyms to make correct judgments and may have also been caused by imperfect judgment of synonymous relationships between Chinese words, which remains a very challenging problem for Chinese.

The entailment relationships between a statement pair may hold for a wide variety of reasons and their combinations, and the organizers of evaluation tasks try to cover as many different types of entailment relationships as possible in the datasets (Dagan et al. 2009; Shima et al. 2012; Watanabe et al. 2013a). As a consequence, the overall performance might not be improved instantly due to the consideration of just one specific factor. Researchers have studied the correlation between datasets and performance of systems (Lin et al. 2015). Hence, it may not be easy to single out and justify the extract contribution of a specific feature with real test data.

The same phenomenon occurred again when we tried to examine the effects of considering syntactic and semantic information to judge entailment relationships with experiments reported in Sect. 4.7.

5.4 World knowledge and subjective judgments

In the real world, we may not be able to judge whether one statement entails another solely by linguistic information

(Vanderwende et al. 2006; Dagan et al. 2009). This is particularly true when world knowledge, connotation and subjective judgments are involved. Following are some statement pairs that were used in NTCIR-10 RITE-2.

Knowledge about the conversion between "米" (meter) and "釐米" (centimeter) is required to judge whether (17) entails (18).

(17) 阿諾爾特大花草直徑能夠到達 3 米
(18) 阿諾爾特大花草直徑 50～90 釐米

The standard answer to the statement pair (19) and (20) is yes, probably because the annotator believed that something that is "最高" (highest) must also be "高" (high). However, this may not be always true, just like the best performer in a contest might not really achieve very high scores.

(19) 鹿茸滋補藥效最高
(20) 鹿茸藥效高

6 Concluding remarks

The main goal of this paper is not to provide a comprehensive survey of studies on textual entailment. Rather, we provide empirical experience obtained from experiments with real test data in NTCIR-9 RITE and NTCIR-10 RITE-2. For additional survey articles that we have not discussed, readers might want to refer to Androutsopoulos and Malakasiotis (2010), Watanabe et al. (2012).

In this paper, we presented the linguistic features and the computational models which we used to achieve second positions in the BC subtask for both simplified and traditional Chinese in NTCIR-10 RITE-2. Significantly extended investigations were carried out, reported, and analyzed to share our empirical experience in textual entailment based on the real data used in NTCIR-9 RITE and NTCIR-10 RITE-2. More experiments, including experiments on English test data used in PASCAL RTE-1 and RTE-2, are available in Huang (2013).

Based on the experience and discussions reported in this paper, we believe that more work on true natural language understanding is needed to achieve better performance in textual entailment recognition. For future work, we are exploring the possibility of applying techniques of textual entailment for answering questions in reading comprehension tests that are designed for language learners (Huang et al. 2013). When computers can do the reading comprehension tests reasonably well, they might also explain the answers to students and serve as a learning companion.

Acknowledgements This research was supported in part by the student travel fund of the Department of Computer Science of National Chengchi University and in part by funding from the Grants of 100-2221-E-004-014, 101-2221-E-004-018, 102-2420-H-001-006-MY2, and 103-2918-I-004-001 of the Ministry of Science and Technology of Taiwan. Access to the digital library services of the Harvard Library was granted to the second author during his visit to Harvard University.

Appendix

We provide information about the Chinese text included in this paper. The *section* column indicates the sections where the Chinese text appears. The *Chinese* column shows the mentioned Chinese text. The *Pronunciation* column shows the pronunciations of the Chinese texts in Hanyu pinyin. The *Translation/Interpretation* column provides a way to interpret the Chinese text in this paper.

Section	Chinese	Pronunciation	Translation/Interpretation
2.1.1	電腦軟體的品質很重要	dian4 nao3 ruan3 ti3 de1 pin3 zhi2 hen3 zhong4 yao4	The quality of computer software is important.
	电脑软体的品质很重要	dian4 nao3 ruan3 ti3 de1 pin3 zhi2 hen3 zhong4 yao4	The quality of computer software is important.
	计算机软件的质量很重要	ji4 suan4 ji1 ruan3 jian4 de1 zhi2 liang4 hen3 zhong4 yao4	The quality of computer software is important.
	计算机	ji4 suan4 ji1	computer
	软件	ruan3 jian4	software
	质量	zhi2 liang4	quality
	工業技術水準	gong1 ye4 ji4 shu4 shui3 zhun3	competence levels of industrial technologies
	工业技术水准	gong1 ye4 ji4 shu4 shui3 zhun3	competence levels of industrial technologies
	工业技术水平	gong1 ye4 ji4 shu4 shui3 ping2	competence levels of industrial technologies
2.1.2	三	san1	three
	參	san1	three
	廿	nian4	twenty
	卅	sa4	thirty
	三十二	san1 shi2 er4	thirty two
	十二	shi2 er4	twelve
	一十二	yi1 shi2 er4	twelve
	朝九晚五	chao1 jiu3 wan3 wu3	work from 9AM to 5PM
	朝 9 晚 5	chao1 jiu3 wan3 wu3	an incorrect way to write 朝九晚五
	舉一反三	ju3 yi1 fan3 san1	providing more examples after being given an example
	舉 1 反 3	ju3 yi1 fan3 san1	an incorrect way to write 舉 1 反 3
2.1.3	研究生命還有多少年	yan2 jiu4 sheng1 ming4 hai2 you3 duo1 shao3 nian2	This string can carry two different meanings, depending on how the Chinese string is segmented. The first meaning is “the number of years remaining for research”, and the second is
			“the remaining life spans of graduate students”.
	研究生命	yan2 jiu4 sheng1 ming4	life for doing research
	還有	hai2 you3	remaining
	多少年	duo1 shao3 nian2	number of years
	研究生	yan2 jiu4 sheng1	graduate students
	命	ming4	Life
2.2.1	Tamara 對於提出這一問題感到猶豫	Tamara dui4 yu2 ti2 chu1 zhe4 yi1 wen4 ti2 gan3 dao4 you2 yu4	Tamara is reluctant to raise this question.
	Tamara 對於詢問這一問題顯得遲疑	Tamara dui4 yu2 xun2 wen4 zhe4 yi1 wen4 ti2 xian3 de1 chi2 yi2	Tamara hesitates to ask this question.
	純化	chun3 hua4	refine
	工	gong1	industrial
	土石	tu3 shi2	stone
	金屬	jin1 shu3	metal
	大	da4	big
	無生物	wu2 shen1 wu4	inanimate
	礦	kuang4	mine
	原	yuan2	original
	冶金	ye3 jin1	metallurgy
	猶豫	you2 yu4	reluctant
	遲疑	chi2 yi2	hesitate
	提出	ti2 chu1	raise
	詢問	xun2 wen4	ask

Section	Chinese	Pronunciation	Translation/Interpretation
2.2.2	無	wu2	not (a character for negation)
	未	wei4	not (a character for negation)
	不	bu4	not (a character for negation)
	非	fei1	not (a character for negation)
	沒有	mei2 you3	not (a word for negation)
	千禧年危機俗稱Ｙ２Ｋ危機或千禧蟲	qian1 xi1 nian2 wei2 ji1 su2 cheng1 Ｙ２Ｋ wei2 ji1 huo4 qian1 xi1 chong2	The problems of year 2000 were commonly called Y2K crisis or thousand-year bugs.
	千禧年危機並非Ｙ２Ｋ危機或千禧蟲	qian1 xi1 nian2 wei2 ji1 bing4 fei1 Ｙ２Ｋ wei2 ji1 huo4 qian1 xi1 chong2	The problems of year 2000 were not called Y2K crisis or thousand-year bugs.
	無法	wu2 fa3	cannot
	未能	wei4 neng2	cannot
	不行	bu4 xing2	cannot
	不能	bu4 neng2	cannot
	無可厚非	wu2 ke3 hou4 fei1	cannot be blamed
3.1.1	千禧年 危機 俗稱 Ｙ２Ｋ 危機或 千禧蟲	qian1 xi1 nian2 wei2 ji1 su2 cheng1 Ｙ２Ｋ wei2 ji1 huo4 qian1 xi1 chong2	The problems of year 2000 were commonly called Y2K crisis or thousand-year bugs.
	千禧年 危機 並非 Ｙ２Ｋ 危機	qian1 xi1 nian2 wei2 ji1 bing4 fei1 Ｙ２Ｋ wei2 ji1 huo4	The problems of year 2000 were not called Y2K crisis or
	或 千禧蟲	qian1 xi1 chong2	thousand-year bugs.
3.1.3	若望保祿二世是四百五十多年來第一位非義大利籍的教宗	ruo4 wang4 bao3 lu4 er4 shi4 shi4 si4 bai3 wu3 shi2 duo1 nian2 lai2 di4 yi1 wei4 fei1 yi4 da4 li4 ji2 de1 jiao4 zong1	Saint John Paul II was the first non-Italian Pope in more than 450 years.
	若望保祿二世是四百五十多年來第一位義大利籍的教宗	ruo4 wang4 bao3 lu4 er4 shi4 shi4 si4 bai3 wu3 shi2 duo1 nian2 lai2 di4 yi1 wei4 yi4 da4 li4 ji2 de1 jiao4 zong1	Saint John Paul II was the first Italian Pope in more than 450 years.
3.1.5	台灣出口至印度成長 28.6%	tai2 wan1 chu1 kou3 zhi4 yin4 du4 cheng2 zhang3 28.6%	Exports from Taiwan to India grew 28.6%.
	印度出口至台灣成長 28.6%	yin4 du4 chu1 kou3 zhi4 tai2 wan1 cheng2 zhang3 28.6%	Exports from India to Taiwan grew 28.6%.
	台灣	tai2 wan1	Taiwan
	印度	yin4 du4	India
3.1.6	台灣出口至印度成長 28.6%	tai2 wan1 chu1 kou3 zhi4 yin4 du4 cheng2 zhang3 28.6%	Exports from Taiwan to India grew 28.6%.
	印度從台灣進口成長率可達 28.6%	yin4 du4 cong2 tai2 wan1 jin4 kou3 cheng2 zhang3 lu4 ke3 da2 28.6%	The growth rate of imports from Taiwan to India can reach 28.6%.
	出口	chu1 kou3	export
	進口	jin4 kou3	Import
	台灣	tai2 wan1	Taiwan
	印度	yin4 du4	India
4.1	香港的主權和領土是在 1997 由英國歸還給中國的	xiang1 gang3 de1 zhu3 quan2 he2 ling3 tu3 shi4 zai4 1997 you2 ying1 guo2 gui1 huan2 gei3 zhong1 guo2 de1	The sovereignty and land of Hong Kong was returned from Britain to China in 1997.
	1997 年香港回歸中國	1997 nian2 xiang1 gang3 hui2 gui1 zong1 guo2	Hong Kong became part of China in 1997.
	一九九一年波斯灣戰爭結束時，雅辛又帶著家人移居約旦	yi1 jiu3 jiu3 yi1 nian2 po1 si1 wan1 zhan4 zheng1 jie2 shu4 shi2，ya3 xin1 you4 dai4 zhe1 jia1 ren2 yi2 ju1 yue1 dan4	When the Gulf War stopped in 1991, Yashin took the family back to Jordan.
	波斯灣戰爭發生於 1991 年	po1 si1 wan1 zhan4 zheng1 fa1 sheng1 yu2 1991 nian2	The Gulf War broke out in 1991.
5.3	聽力	ting1 li4	ability to hear
	聽覺	ting1 jue2	hearing perception

Section	Chinese	Pronunciation	Translation/Interpretation
	噪聲對動物也有很大的影響，降低動物聽力	zao4 sheng1 dui4 dong4 wu4 ye3 you3 hen3 da4 de1 ying3 xiang3，jiang4 di1 dong4 wu4 ting1 li4	Environmental noise impacts animals' hearing as well, reducing their hearing ability.
	噪聲對動物的聽覺有很大的影響	zao4 sheng1 dui4 dong4 wu4 de1 ting1 jue2 you3 hen3 da4 de1 ying3 xiang3	Environmental noise has a great influence on animals' hearing ability.
	阿諾爾特大花草直徑能夠到達 3 米	a1 nuo4 er3 te4 da4 hua1 cao3 zhi2 jing4 neng2 gou4 dao4 da2 3 mi3	The diameter of Rafflesia arnoldii can reach 3 meters.
	阿諾爾特大花草直徑 50～90 釐米	a1 nuo4 er3 te4 da4 hua1 cao3 zhi2 jing4 50~90 li2 mi3	The diameter of Rafflesia arnoldii is between 50 and 90 centimeters.
5.4	鹿茸滋補藥效最高	lu4 rong2 zi1 bu3 yao4 xiao4 zui4 gao1	The medical effects of velvet antler are the highest.
	鹿茸藥效高	lu4 rong2 yao4 xiao4 gao4	The medical effects of velvet antler are high.

References

Androutsopoulos I, Malakasiotis P (2010) A survey of paraphrasing and textual entailment methods. J Artif Intell Res 38:135–187

Bar-Haim R, Dagan I, Dolan B, Ferro L, Giampiccolo D, Magnini B (2006) The second PASCAL recognising textual entailment challenge. In: Proceedings of the second PASCAL challenges workshop on recognising textual entailment

Bar-Haim R, Dagan I, Mirkin S, Shnarch E, Szpektor I, Berant J, Greenthal I (2008) Efficient semantic deduction and approximate matching over compact parse forests. In: Proceedings of the TAC 2008 workshop on textual entailment

Budanitsky A, Hirst G (2006) Evaluating WordNet-based measures of lexical semantic relatedness. Comput Linguist 32(1):13–47

Burchardt A, Pennacchiotti M, Thater S, Pinkal M (2009) Assessing the impact of frame semantics on textual entailment. Nat Lang Eng 15(4):527–550

Chambers N, Cer D, Grenager T, Hall D, Kiddon C, MacCartney B, de Marneffe MC, Ramage D, Yeh E, Manning CD (2007) Learning alignments and leveraging natural logic. In: Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing, pp 165–170

Chang CC, Lin CJ (2011) LIBSVM: a library for support vector machines. In: ACM Trans Intell Syst Technol 2(3):article 27

Chang PC, Galley M, Manning CD (2008) Optimizing Chinese word segmentation for machine translation performance. In: Proceedings of the third workshop on statistical machine translation, pp 224–232

Chang PC, Tseng H, Jurafsky D, Manning CD (2009) Discriminative reordering with Chinese grammatical relations features. In: Proceedings of the third workshop on syntax and structure in statistical translation

Chang TH, Hsu YC, Chang CW, Hsu YC, Chang JI (2013) KC99: a prediction system for Chinese textual entailment relation using decision tree. In: Proceedings of the tenth NTCIR conference, pp 469–473

Chen KJ (2013) Lexical semantic representation and semantic composition: an introduction to E-HowNet. http://rocling.iis.sinica.edu.tw/CKIP/paper/Technical_Reprt_E-HowNet.pdf

Chen WT, Lin SC, Huang SL, Chung YS, Chen KJ (2010) E-HowNet and automatic construction of a lexical ontology. In: Proceedings of twenty-third international conference on computational linguistics (demonstration volume), pp 45–48

Chuang YH, Liu CL, Chang JS (2012) Effects of combining bilingual and collocational information on translation of English and Chinese verb-noun pairs. Int J Comput Linguist Chin Lang Process 17(3):1–28

Dagan I, Dolan B, Magnini B, Roth D (2009) Recognizing textual entailment: rational, evaluation and approaches. Nat Lang Eng 15(4):i–xvii

Dagan I, Glickman O, Magnini B (2006) The PASCAL recognising textual entailment challenge. Lect Notes Comput Sci 3944:177–190

Day MY, Tu C, Huang SJ, Vong HC, Wu SW (2013) IMTKU textual entailment system for recognizing inference in text at NTCIR-10 RITE2. In: Proceedings of the tenth NTCIR conference, pp 462–468

de Salvo Braz R, Girju R, Punyakanok V, Roth D, Sammons M (2005) Knowledge representation for semantic entailment and question-answering. In: Proceedings of IJCAI-05 workshop on knowledge and reasoning for question answering

Duan H, Sui Z, Tian Y, Li W (2012) The CIPS_SIGHAN CLP 2012 Chinese word segmentation on microblog corpora bakeoff. In: Proceedings of the second CIPS-SIGHAN joint conference on Chinese language processing, pp 35–40

Fillmore CJ (1976) Frame semantics and the nature of language. Ann N Y Acad Sci 280(1):20–32

Firth JR (1935) The technique of semantics. Trans Philolog Soc 34(1):36–73

Firth JR (1957) A synopsis of linguistic theory 1930–1955. In: Studies in linguistic analysis, pp 1–32

Gao J, Li M, Wu A, Huang CN (2005) Chinese word segmentation and named entity recognition: a pragmatic approach. Comput Linguist 31(4):531–574

Harris Z (1954) Distributional structure. Word 10(23):146–162

Huang WJ (2013) Textual Entailment Recognition for Chinese and English. Master's Thesis, Department of Computer Science, National Chengchi University, Taiwan

Huang WJ, Lin PC, Liu CL (2013) An exploration of textual entailment and reading comprehension for Chinese and English. In: Proceedings of the twenty-fifth conference on research on computational linguistics and speech processing, pp 105–119

Huang WJ, Liu CL (2013) NCCU-MIG at NTCIR-10: using lexical, syntactic, and semantic features for the RITE tasks. In: Proceedings of the tenth NTCIR conference, pp 430–434

Levy R, Manning CD (2003) Is it harder to parse Chinese, or the Chinese Treebank? In: Proceedings of the forty-first annual meetings of association for computational linguistics, pp 439–446

Liu CL, Pai TW (2006) Methods for path and service planning under route constraints. Int J Comput Appl Technol 25(1):40–49

Lin CJ, Lee CW, Shih CW, Hsu WL (2015) Rank correlation analysis of RITE datasets and evaluation metrics—an observation on NTCIR-10 RITE Chinese subtasks. Web Intell 13(2)

Lloret E, Ferrández Ó, Muñoz R, Palomar M (2008) A text summarization approach under the influence of textual entailment. In: Proceedings of the fifth international workshop on natural language processing and cognitive science, pp 22–31

Nielsen RD, Ward W, Martin JH (2009) Recognizing entailment in intelligent tutoring systems. Nat Lang Eng 15(4):479–502

Page L, Brin S, Motwani R, Winograd T (1998) The Pagerank citation ranking: bringing order to the web. Technical report, Stanford Digital Library Technologies Project

Shibata T, Kurohashi S, Kohama S, Yamamoto A (2013) Predicate-argument structure based textual entailment recognition system of Kyoto team for NTCIR-10 RITE-2. In: Proceedings of the ninth NTCIR conference, pp 537–544

Shih CW, Liu C, Lee CW, Hsu WL (2013) IASL RITE system at NTCIR-10. In: Proceedings of the tenth NTCIR conference, pp 425–429

Shima H, Kanayama H, Lee CW, Lin CJ, Mitamura T, Miyao Y, Shi S, Takeda K (2012) Overview of NTCIR-9 RITE: recognizing inference in text. In: Proceedings of the ninth NTCIR conference, pp 291–301

Stern A, Lotan A, Mirkin S, Shnarch E, Kotlerman L, Berant J, Dagan I (2011) Knowledge and tree-edits in learnable entailment proofs. In: Proceedings of the text analysis conference (TAC'11)

Stern A, Shnarch E, Lotan A, Mirkin S, Kotlerman L, Zeichner N, Berant J, Dagan I (2010) Rule chaining and approximate match in textual inference. In: Proceedings of the text analysis conference (TAC'10)

Takesue Y, Ninomiya T (2013) EHIME textual entailment system using Markov logic in NTCIR-10 RITE-2. In: Proceedings of the tenth NTCIR conference, pp 507–511

Tatar D, Mihis AD, Lupsa D, Tamaianu-Morita E (2009) Entailment-based linear segmentation in summarization. Int J Softw Eng Knowl Eng 19(8):1023–1038

Tsujii J (2012) Natural language understanding, semantic-based information retrieval and knowledge management. In: Proceedings of the ninth NTCIR conference, p 8

Vanderwende L, Menezes A, Snow R (2006) Microsoft research at RTE-2: syntactic contributions in the entailment task: an implementation. In: Proceedings of the second PASCAL challenges workshop on recognising textual entailment

Wang XL, Zhao H, Lu BL (2013) BCMI-NLP labeled-alignment-based entailment system for NTCIR-10 RITE-2 task. In: Proceedings of the tenth NTCIR conference, pp 474–478

Watanabe Y, Miyao Y, Mizuno J, Shibata T, Kanayama H, Lee CW, Lin CJ, Shi S, Mitamura T, Kando N, Shima H, Takeda K (2013a) Overview of the recognizing inference in text (RITE-2) at NTCIR-10. In: Proceedings of the tenth NTCIR conference, pp 385–404

Watanabe Y, Mizuno J, Inui K (2013b) THN's natural logic-based compositional textual entailment model at NTCIR-10 RITE-2. In: Proceedings of the tenth NTCIR conference, pp 531–536

Watanabe Y, Mizuno J, Nichols E, Narisawa K, Nabeshima K, Okazaki N, Inui K (2012) Leveraging diverse lexical resources for textual entailment recognition. ACM Trans Asian Lang Inf Process 11(4):Article 18

Witten IH, Frank E, Hall MA (2011) Data mining: practical machine learning tools and techniques. Morgan Kaufmann, Burlington

Wu SH, Yang SS, Chen LP, Chiu HS, Yang RD (2013) CYUT Chinese textual entailment recognition system for NTCIR-10 RITE-2. In: Proceedings of the tenth NTCIR conference, pp 443–448

Yarowsky D (1995) Unsupervised word sense disambiguation rivaling supervised methods. In: Proceedings of the thirty-third annual meeting of the association for computational linguistics, pp 189–196