Context-based Translation of Constant Concept Values in E-Business

Xin Guan and Jingzhi Guo
Department of Computer and Information Science, University of Macau
Av. Padre Tomás, Pereira, S.J., Taipa, Macau, Tel: +853-397 4296
E-mail:{ma46601, jzguo}@umac.mo

The heterogeneous concept mapping (HCM) approach of COllaborative CONcept EXchange (CONEX) supports the multilingual business information exchange between heterogeneous e-business systems, but it unexplored the issue of semantic consistency maintenance in the constant value translation. This paper discusses this issue with a proposal of Contextual vAlue Translation (CAT) approach to complementing the previous HCM research. This approach provides the solution to facilitate the multilingual translation of business concept values with semantic consensus.

I. Introduction

In real-world e-business environment, business information exchange is often not limited to a single natural language environment. Due to the characteristic of cross-border international trade, e-commerce transactions are often made between different natural languages. An early research of heterogeneous concept mapping (HCM) for COllaborative CONcept EXchange (CONEX) [6] supports the multilingual business information exchange between heterogeneous e-business information systems. Nevertheless, the translation of constant concept values between different natural languages has not been extensively explored in the HCM of CONEX. Technically speaking, HCM did not touch the issue of semantic consistency maintenance on the constant values of concepts between different natural languages. The *constant value of a concept* is defined as a concrete or conceptual entity instance of a concept in a given context of a natural language, for instance, "red" is a constant value of the "color" concept. We illustrate this issue with two pieces of product information (e.g. refrigerator) in English (say PI_1) and Chinese (say PI_2) shown in Fig. 1, where both contain the concepts with meanings and their corresponding constant values of concepts.

```
<domesticrefrigerator>
   <clr>silver</clr>
   <place> CN</place>
   <brand>Haier</brand>
</domesticrefrigerator>
```

PI_1

家用冰箱	产地	品牌	颜色
	中国	海尔	银

PI_2

Fig. 1. Two pieces of product information

Given PI_1 and PI_2, one of the research goals of HCM [6] is to semantically transform PI_1 into a piece of information that can be understood by the recipients in the context of PI_2, such that [a] the concepts of "domestic refrigerator", "clr", "place" and "brand" that are not concrete concept instances in PI_1 context can be transformed into the concepts of "家用冰箱", "颜色", "产地" and "品牌" in PI_2 context in a semantically consistent way, and [b] the constant concept values of "silver", "CN" and "Haier" that are concrete concept instances in PI_1 context can be correctly translated into the semantically consistent Chinese concept values in PI_2 context. In [6] a heterogeneous concept transformation algorithm was designed to fulfill the above required concept transformation. The proposed algorithm can successfully transform heterogeneous concepts from PI_1 context to PI_2 context (i.e. [a] problem solved), but have not further investigated the behaviors of the accurate translation of constant concept values of PI_1 concepts into those of PI_2 concepts (i.e. [b] problem remained unsolved).

The aim of this paper is to enable the accurate translation of constant concept values to solve the above [b] problem, i.e. the semantic consistency maintenance on constant value translation between different natural languages, through a novel solution called *Contextual vAlue Translation* (CAT) framework. With this solution, we particularly make contributions to the following:

- Integrate CONEX context information with term sense disambiguation (TSD) [10] to achieve semantic consistency of constant values
- Improve the accuracy of constant value translation by CAT framework.

The remainder of this paper arranges as follows: Section 2 investigates some useful related work for building CAT framework. Section 3 describes our novel CAT framework. In Section 4, we discuss some of the important features of CAT framework and compare them with the other existing translation systems and lexical resources. Finally, we conclude our paper by proposing the future work.

II. Related Work

A. Term Sense Disambiguation vs. Word Sense Disambiguation

Recently, many lexical resources, such as (1) WordNet [14], MindNet [11], FrameNet [2] by America, (2) the conceptual dictionary EDR [1] by Japan, and (3) the HowNet [7], CCD [9] and SKCC [8] by China, play essential important roles for the Word Sense Disambiguation (WSD) [13] in Natural Language Processing. These lexical resources for WSD application are fairly successful. However, there are several reasons that these lexical resources for WSD application are not entirely suitable for our work. Firstly, some business terms are domain-specific,

1-4244-1007-X/07/$25.00 ©2007 IEEE.

which means that general language resources and techniques may not be appropriate [10]. Secondly, some constant values as *special terms,* which are not covered sufficiently in any general language dictionary or thesaurus. Thirdly, the ambiguity of multiword term (some constant value) is generally caused not by different senses of the individual components of a term, but by different senses of a term as a whole [10]. Therefore, the general lexical resources may not be appropriate for term sense disambiguation (TSD) [10] application.

Due to the limitations of the general lexical resources for the constant values supplying and TSD application, we propose the domain-specific dictionary which contains abundant specialized constant value terms, and in which the TSD application can be fulfilled.

B. CONEX Context Analysis

In heterogeneous concept mapping research of CONEX, the context is “the semantics definition of a product related to local product representations in a semantic community and common product representations in a product catalogue provider” [5]. A CONEX context framework can be described in Fig. 2, which shows how heterogeneous contexts are mapped and transformed. The PI$_1$ as an active context is transformed into PI$_2$ context through a context transformation chain: active context PI$_1$→local context LEPC1 $\xrightarrow{\text{LCMAP1}}$ common context CEPC1 $\xrightarrow{\text{CCMAP1}}$ common context CEPC2 $\xrightarrow{\text{CLMAP2}}$local context LEPC2 →PI$_2$ context.

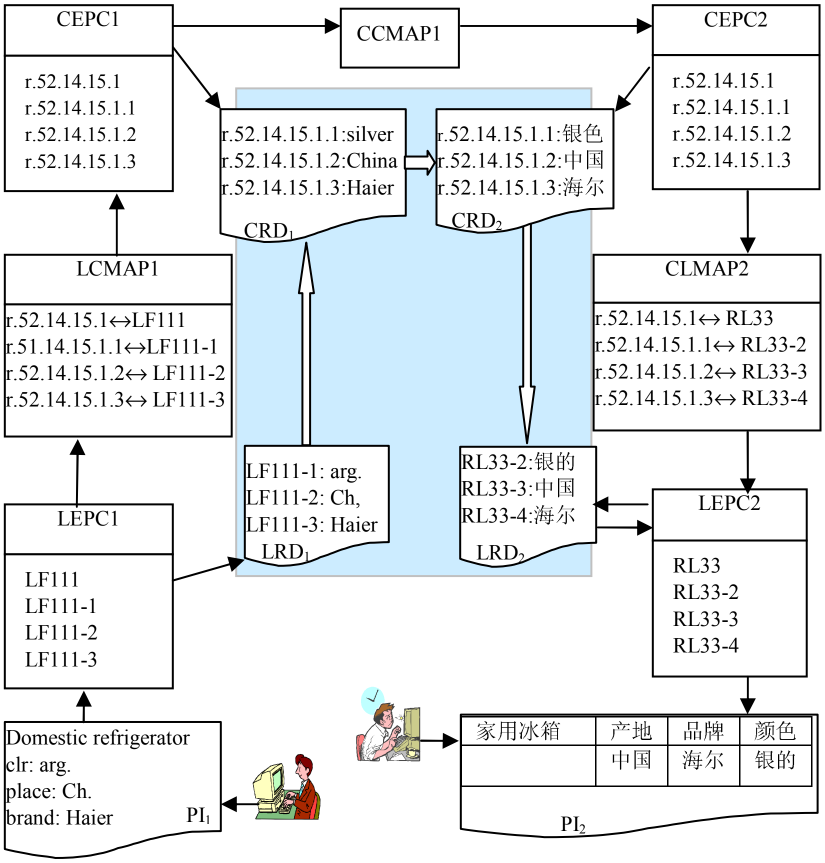

Fig. 2. CONEX heterogeneous context transformation and concept mapping

In this paper, we choose the CONEX product concept as the context of product constant values. For each constant value, its associating context derives from the leaf product concept. Take a piece of reified product document for example:

```
<c iid="r.52.14.15.1.1" an="color" g="Constant" co="0">
      <val dt="string"> silver</val></c>
```

In the example, constant value “silver” has a context “color”, which is given by the leaf concept c(r.52.14.15.1, color, constant, 0). The leaf product concept identifier iid r.52.14.15.1.2 identifies a specific color context of refrigerator of domestic appliances. It provides an accurate context recursive definition for interpreting “silver”. With this accurate context, “silver” can be accurately translated. CONEX context analysis approach performs the TSD task to solve the ambiguous homographs issue.

III. CAT Framework

The novel CAT framework for constant value translation between two different natural language contexts is a 2-component framework <(*LC, CC*), (*LC-TE, CC-TE,CL-TE*)> including a dictionary component (LC, CC) and a translation engine component (LC-TE, CC-TE, CL-TE). *LC* refers to a labeled multi-set, where each set is *Local Contextual vAlue Dictionary* (LCAD). *CC* refers to a labeled multi-set, where each set is *Common Contextual vAlue Dictionary* (CCAD). *LC-TE* is *local-Common translation engine* for transforming local constant values to common constant values through LCAD. *CC-TE* is common-common translation engine for translating common constant values of one language into those of another language through CCAD. *CL-TE* is *common-local translation engine* for transforming common constant values to local constant values through LCAD. The general architecture of CAT Framework can be illustrated in Fig. 3 to explain its work. In Fig. 3, the top layer is dictionary design layer. The bottom layer is constant value translation layer. Both Layers follow the design principles of HCM mechanism of CONEX [6].

A. CAT Design Layer

In the design layer of Fig. 3, designers are divided into two categories: *Common Contextual vAlue Dictionary Designers* (CCADD) and *Local Contextual vAlue Dictionary Designers* (LCADD). With this in mind, we have:

- CCADD are responsible for collaborative creation and editing common constant values that can be used in all CCADs with different languages.
- LCADD, any LCAD designer, which collaborates with CCADD to form a D2F (dominant-to-follower) collaboration community [4]. The result of D2F collaboration is that LCADD has personalized the common constant values into the local constant values

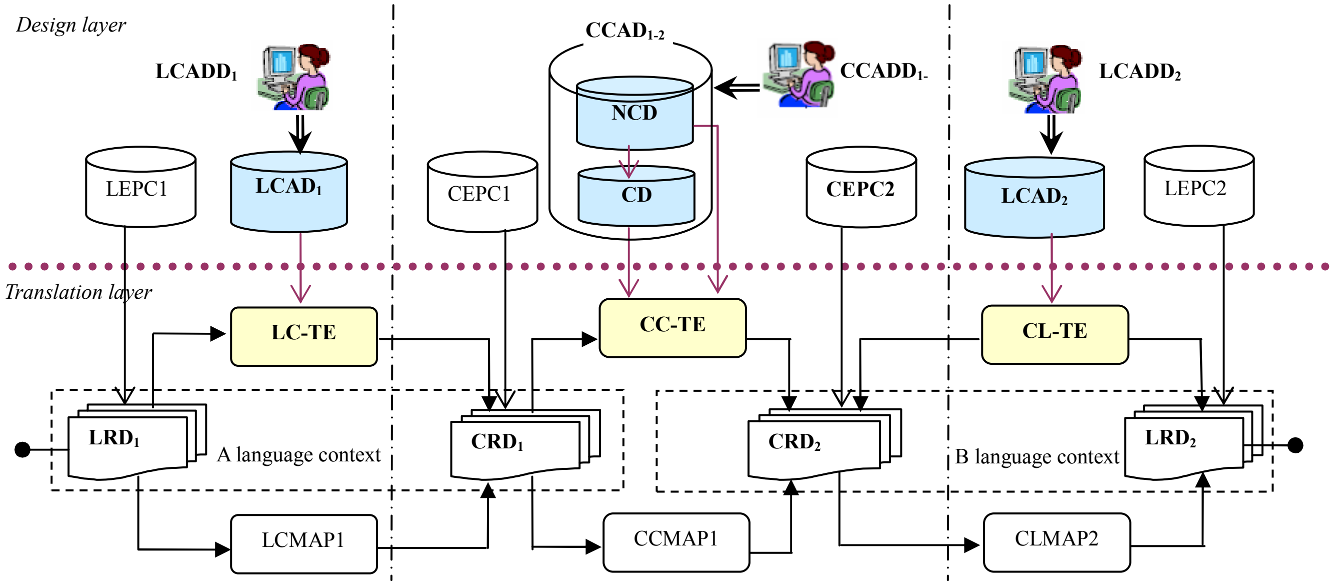

Fig. 3. The architecture of CAT

such that LCAD is created and edited.

- CCAD, any *bilingual contextual value dictionary*, is a couple <CD, NCD>, which consists of two sub-dictionaries: *context dictionary* (CD) and *non-context dictionary* (NCD). In CD, it includes a large volume of context information that is used for providing the accurate interpretation of constant values. The structure of CD is represented in a 3-tuple:

$$CD = <CI, RI, CV>$$

CI refers to a set of *common identifiers* (ComIids), where each ComIid has represented the full semantic of a common concept on CEPC and is qualified to replace a common concept. RI refers to an *abstract concept* (e.g. color, brand, etc…) and is the concept definition of CI, where each *abstract concept* creation is depended on the CCADD. CV refers to a set of *common constant values* in context RI, where each value is selected as a preferred term by CCADD collaboration. An example of CD structure is shown in the following Fig. 4.

$$CD = \begin{array}{l} \left\langle \left\{ \begin{array}{c} r.1.1.1.1 \\ r.52.14.15.1.1 \\ \ldots \end{array} \right\}, color, \left\{ \begin{array}{l} Silver \\ light\ \ blue \end{array} \right\} \right\rangle \\ \left\langle \left\{ \begin{array}{c} r.1.1.1.2 \\ r.52.14.15.1.3 \\ \ldots \end{array} \right\}, brand, \left\{ \begin{array}{c} Haier \\ Dell \\ \ldots \end{array} \right\} \right\rangle \\ \left\langle \left\{ \begin{array}{c} r.1.1.1.3 \\ r.52.14.15.1.2 \\ \ldots \end{array} \right\}, place, \left\{ \begin{array}{c} Japan \\ China \\ \ldots \end{array} \right\} \right\rangle \\ \quad \ldots\ldots \end{array}$$

Fig. 4. An example of CD structure

Different from CD, NCD is designed to establish the corresponding relationship without any context information between two different language common constant values. The structure of NCD is a 2-tuple:

$$NCD= <S\text{-}CV, T\text{-}CV >$$

An example of NCD can be seen in Fig. 5, where S-CV refers to the source language common constant value, and the T-CV refers to the corresponding target language common constant value. An example of NCD structure is shown in Fig. 5.

$$NCD = \begin{array}{l} China \rightarrow \text{中国} \\ Haier \rightarrow \text{海尔} \\ silver \rightarrow \left\{ \begin{array}{c} \text{银色} \\ \text{银子} \\ \ldots \end{array} \right\} \\ \ldots\ \ \ldots \end{array}$$

Fig. 5. An example of NCD structure

- LCAD, any *personalized dictionary* which includes the context information for interpreting both common and local constant values. In the dictionary, the equivalence relationship between common constant values and local constant values are clearly represented. The structure of LCAD is a 3-tuple:

$$LCAD = <RI:CV, M, LI:LV >$$

RI:CV refers to couple with RI as context and one candidate in CV as contextual candidate common constant value. LI:LV refers to a set of 2-componnent structure with LI local identifiers (LocIids) and LV (local candidate constant values), where each LI corresponds to one possible local constant value. M is a map, functioning map RI:CV and LI:LV if two conditions are satisfied such that (1) one candidate CVi and one candidate LVi have the same or similar (but replaceable) semantics, and (2) The RI of CVi is concept-equivalent to LocIid of LVi. An example of LCAD structure is shown in Fig. 6.

$$\text{LCAD} = \begin{array}{l} \left\langle \text{color} : \text{silver} \leftrightarrow \begin{Bmatrix} \text{LF111-1} : \text{arg.} \\ \text{MF122-1} : \text{argent} \\ \ldots \end{Bmatrix} \right\rangle \\ \left\langle \text{place} : \text{China} \leftrightarrow \begin{Bmatrix} \text{LF111-2} : \text{CN} \\ \text{MF122-2} : \text{Chin.} \\ \ldots \end{Bmatrix} \right\rangle \\ \qquad \ldots \ldots \end{array}$$

Fig. 6. An example of LCAD structure

B. CAT Translation Layer

In the translation layer of Fig. 3, the complete translation process is based on the CONEX context transformation [6]. Therefore, some research results of CONEX, such as common context, concept storage and concept supply chain, are directly used in this paper (readers are referred to [6]).

In this paper, the process of constant value translation is achieved by three procedures: (1) LC-TE is responsible for Local-to-Common constant value transformation, (2) CC-TE is responsible for Common-to-Common constant value translation, and (3) LC-TE is responsible for common-local constant value transformation.

Procedure 1: The *Local-Common Translation Engine* (LC-TE) is to transform the local constant values in LRD$_1$ into the common constant values in CRD$_1$ based on LCAD$_1$. Core to this transformation is that LC-TE compares whether the local constant values of LRD$_1$ is in the LCAD$_1$. If so, then LRD$_1$ can be transformed into CRD$_1$. Given a LCAD$_1$, LC-TE transforms the LRD$_1$ to CRD$_1$ if:

(1) $\forall LI \in LRD_1$

(2) $\forall$ $LI{:}LV \wedge RI{:}CV \in M$ of $LCAD_1 \bullet map(RI{:}CV, LI{:}LV)$

The first condition suggests that LI (LocIid) of LEPC$_1$ must fall in the LRD$_1$. The second condition suggests that the (LI) LocLid and its associating local constant value of LRD$_1$ must fall in the LCAD$_1$.

We give an example for explaining this transformation procedure shown in Fig. 7. In this example, given a piece of LRD$_1$ "LF111-1: arg.", according to LF111-1, LC-TE gets the RI context information "color" in LCAD$_1$. And then, LCAD$_1$, "LF111-1: arg." and 'color' can identify one item with CV 'silver'. Meantime, the local concept "LF111-1" can be transformed into "r.52.14.15.1.1" by CONEX transformer through LCMAP1 [6]. Finally, the piece of CRD$_1$ "r.52.14.15.1.1: silver" is obtained.

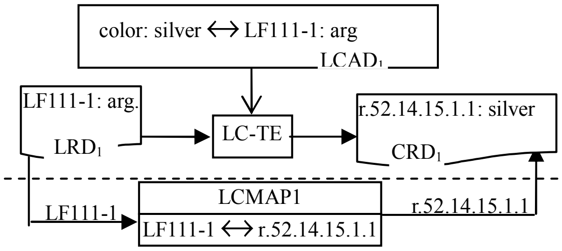

Fig. 7. An example of procedure 1

Procedure 2: The *Common-Common Translation Engine* (CC-TE) is to translate the common constant values in CRD$_1$ of *A* language context into those of *B* language context through CCAD$_{1\text{-}2}$. Given a CCAD$_{1\text{-}2}$, CC-TE translates CRD$_1$ into CRD$_2$ if:

(1): $S\text{-}CV \in NCD$

(2): $\forall ComIid \in CRD_1$

(3): $\forall T\text{-}CV \in NCD$; CI, RI and $T\text{-}CV \in CD$

Condition (1) states that the common constant values of source language(S-CV) must fall in the NCD. Condition (2) states that the ComLid must fall in the CRD$_1$. Condition (3) states that the common constant values of target language (T-CV) must fall in the NCD. The ComIid (CI), context information (RI) and the common constant values of target language (T-CV) must fall in the CD.

We give an example for explaining this translation procedure shown in Fig. 8.

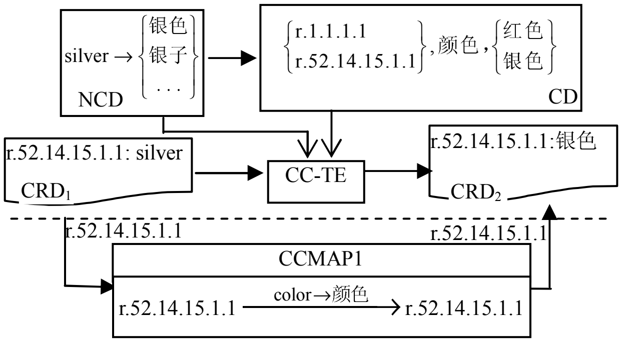

Fig. 8. An example of procedure 2

In this example, given a piece of "r.52.14.15.1.1: silver" in CRD$_1$, according to NCD, the English term "silver" has two corresponding Chinese common constant values which are T-CV: "银色" and "银子". And then according to the "r.52.14.15.1.1" and "颜色" in CD, CV with {红色, 银色, ...} can be obtained. Finally, the correct target Chinese common constant value "银色" is obtained by {**银色**, 银子}∩{红色, **银色**...}. Meantime, the common concept "r.52.14.15.1.1" $\leftarrow$color in *A* language context can be replicated into "r.52.14.15.1.1" $\leftarrow$颜色 in *B* language context through CCMAP1 [6]. Finally, the piece of CRD$_2$ "r.52.14.15.1.1: 银色"is obtained. One condition should be mentioned, that is, in NCD, if the common constant values of source language corresponds to only one common constant values of target language, the CC-TE directly get the correct target common constant values through NCD and the CD is not necessary. In other word, the CD is mostly utilized to disambiguate ambiguous polysemous constant values.

Procedure 3: The *Common-Local Translation Engine* (CL-TE) is to transform the constant values in CRD$_2$ into the local constant values in LRD$_2$ based on LCAD$_2$. Given a LCAD$_2$, CL-TE transforms the CRD$_2$ into LRD$_2$ if:

(1) :$\forall CI \in CRD_1$

(2): $\forall$ $LI{:}LV \wedge RI{:}CV \in M$ of $LCAD_2 \bullet map(RI{:}CV, LI{:}LV)$

The first condition suggests that CI (ComIid) of CEPC$_2$ must fall in the CRD$_2$. The second condition suggests that the CI and its associating common constant value of CRD$_2$ must fall in the LCAD$_2$.

We give an example for explaining this transformation procedure shown in Fig. 9.

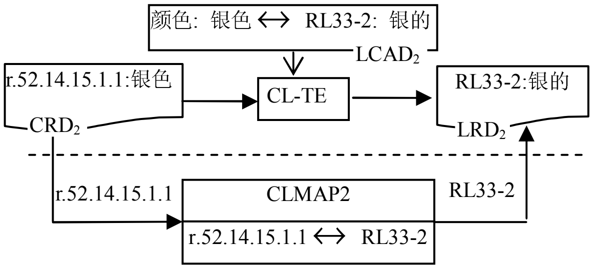

Fig. 9. An example of procedure 3

In this example, given a piece of "r.52.14.15.1.1: 银色" in CRD$_2$, according to "r.52.14.15.1.1", the RI is '颜色'. And then, in the LCAD$_2$ of CRD$_2$, "r.52.14.15.1.1:银色" and 'color' can identify one item with LV '银的'. Meantime, the common concept "r.52.14.15.1.1" can be transformed into "RL33-2" by CONEX transformer through CLMAP2 [6]. Finally, the piece of LRD$_2$ "RL33-2: 银的" is obtained.

IV. Discussion and Comparison

CAT approach resolves the issue of semantic consistency maintenance in the constant value translation. This approach presents a CAT framework which consists of the dictionary design layer and the translation layer. Within the two layers, the CAT approach has several features:

(1) Context-based. The semantics of each constant value in LCAD and CD are interpreted by the CONEX concept context, which enables the constant value translation.

(2) Accurate translation: CAT approach performs the term sense disambiguation to correctly translate the constant values without any semantic ambiguity.

(3) Extensibility: Any local designer can design personalized LCAD easily and efficiently to integrate into CAT framework without any change of the local design.

We compare the features of CAT with some existing translation systems (e.g. SYSTRAN [12] and EDR [1]) and lexical resources (e.g. WordNet [14] and SKCC [8]), and the result is shown in Table I.

TABLE I

Feature Comparison of Existing Translation Systems and Lexical Resources

Features	**SYSTRAN**	**EDR**	**WordNet**	**SKCC**
Context based	Not support	Support	Support	Support
Accurate translation	Low	Medium	Not support	Not support
Extensibility	Medium	Low	High	Low

V. Conclusion

This paper has proposed a novel CAT approach. Its contribution is its new support of multilingual translation of business concept values with semantic consensus. A future work of this paper is to refine the *Contextual vAlue Dictionary* design for constant value translation.

Acknowledgement

The work reported in this paper has been partially supported by University of Macau Research Grand.

References

[1] EDR, Japan Electronic Dictionary Research Institute, Ltd.: http://www.iijnet.or.jp/edr, accessed on 25th, June.

[2] FrameNet, an on-line lexical resource for English, http://framenet.icsi.berkeley.edu/, accessed on 25th, June.

[3] Guo. J (2006) "Achieving Transparent Integration of Information, Documents and Processes", in: Proc. of IEEE CEBE'06, IEEE Computer Society Press, pp. 559-562.

[4] Guo, J. (2006) "A Transparent Collaborative Integration Approach for Ad Hoc Product Data", in: Proc. of CEC/EEE'07, IEEE Computer Society Press.

[5] Guo, J. and Sun, J. (2003) "Context Representation of Product Data", ACM SIGEcom Exchanges 4(1), pp. 20-28.

[6] Guo, J., Sun, C. and Chen, D. (2004) "Transforming Heterogeneous Product Concepts through Mapping Structures", in: Proc. of the 2004 Int'l Conf. on Cyberworlds (CW'04), IEEE Computer Society, pp. 22-29.

[7] HowNet, an online common-sense knowledge base, http://www.keenage.com, accessed on 25th, June.

[8] Hui Wang and Shiwen Yu (2003) "The Semantic Knowledge-base of Contemporary Chinese and its Applications in WSD", in: Proc. of the second SIGHAN workshop on Chinese language processing. pp. 112-118.

[9] Liu, Y. Yu, S. W. and Yu, J. S. (2002) "Building a Bilingual WordNet-Like Lexicon: the New Approach and Algorithms", in: Proc. Of COLING'02, Taipei, China.

[10] Maynard, D. and S. Ananiadou (1998) "Term sense disambiguation using a domain-specific thesaurus", in: Proc. of 1st Int'l Conf. on Language Resources and Evaluation (LREC), Granada, Spain.

[11] MindNet, an automatically-created lexical resource, http://reserach.microsoft.com/nlp/Projects/MindNet.aspx, accessed on 25th, June.

[12] SYSTRAN, a language translation software, http://systransoft.com, accessed on 25th, June.

[13] Vickrey, D., Biewald, L., Teyssier, M. and D. Koller (2005) "Word-sense disambiguation for machine translation", in: Proc. of Conf. on Human Language Technology and Empirical Methods in Natural Language, ACM Press, pp. 771-778.

[14] WordNet, a lexical database for the English language, http://en.wikipedia.org/wiki/Wordnet, accessed on 25th, June.