The 2018 Conference on Computational Linguistics and Speech Processing
ROCLING 2018, pp. 256-265
©The Association for Computational Linguistics and Chinese Language Processing

Smart vs. Solid Solutions in Computational Linguistics
—Machine Translation or Information Retrieval

Su-Mei Shiue
Department of Computer Science and Information Engineering
National Taipei University of Technology
t102599005@ntut.edu.tw

Lang-Jyi Huang
Department of Computer Science and Information Engineering
National Taipei University of Technology
t8599003@ntut.edu.tw

Wei-Ho Tsai
Department of Electronic Engineering
National Taipei University of Technology
whtsai@ntut.edu.tw

Yen-Lin Chen
Department of Computer Science and Information Engineering
National Taipei University of Technology
ylchen@csie.ntut.edu.tw

Abstract

Smart solutions and solid solutions are two different approaches for computational linguistics. To receive a satisfactory result at a minimum cost, smart solutions use special properties of the on-hand application. Solid solutions, in contrast, more generally analyze the application in order that these solutions can be used for a whole class of applications. By extending a well-maintained solid solution, it may be re-used once and again, thus saving time, money, and other resources in the long run. Therefore, the research focuses on solid solutions. It To implement smart solutions, machine translation is used while information retrieval is for solid solutions.

Keywords: Smart Solutions, Solid Solutions, Computational Linguistics, Grammatical

Components, Natural Language.

1. Introduction

1.1 Destination of Computational Linguistics - Natural Transmission of Information

The goal of computational linguistics is to reproduce the natural transmission of information by modeling the speaker's production and the hearer's interpretation on a suitable type of computing machine. This amounts to the construction of autonomous cognitive machines (robots) which can freely communicate in natural language. And the development of speaking robots is a real scientific task [1].

1.2 Turing Test - Modeling the Mechanism of Natural Communication

The task of modeling the mechanism of natural communication on the computer was described in 1950 by [4] in the form of an 'imitation game' known today as the Turing test The computer passes the Turing test if the man or the woman replaced by the computer is simulated so well that the guesses of the human interrogator are just as often right and wrong as with the earlier natural partner. In this way Turing wanted to replace the question "Can machines think?" with the question "Are there imaginable digital computers which would do well in the imitation game?" [1, 10]

1.3 Eliza Program – A Prototype of Smart Solution

The Eliza program [2] simulates a psychiatrist encouraging the human interrogator, now in the role of a patient, to talk more and more about him- or herself. Eliza works with a list of words. Whenever one of these words is typed by the human interrogator/patient, Eliza inserts it into one of several prefabricated sentence templates. For example, when the word mother is used by the human, Eliza uses the template Tell me more about your ___ to generate the sentence Tell me more about your mother. Eliza program is the prototype of a smart solution in that it exploits the restrictions of a highly specialized application to achieve a maximal effect with a minimum of effort.

2. Smart versus Solid Solutions

For computational applications, two distinguished approaches are [1]: smart solutions and solid solutions. Smart solutions use special properties of the particular application to obtain a sufficient result at a minimum of cost. Solid solutions, on the other hand, analyze the application at a more general level such that they can be used for a whole class of applications. By maintaining and extending a solid solution, it may be re-used again and again, thus saving time and money in the long run.

3. Solid Solutions in Computational Applications [1]

3.1. Indexing and Retrieval in Textual Databases

The electronic index is built up automatically when the texts are read into the database, whereby the size of the index is roughly equal to that of the textual database itself. The use of an electronic index has the following advantages over a card index: power of search, flexibility, general specification of patterns (general specification of patterns, combination of patterns), automatic creation of the index structure, ease, speed, and reliability

The advantages of electronic search apply to both the *query* (input of the search words) and the *retrieval* (output of the corresponding texts or passages).

3.2 Grammatical Knowledge Improves Retrieval from Textual Databases.

3,2.1 Phenomena Requiring Linguistic Solutions

The reason for the surprisingly low recall of only 20 % on average is that STAIRS uses only technological, i.e., letter-based, methods [5]. Using grammatical knowledge in addition, recall could be improved considerably. Textual phenomena which resist a technological treatment, but are suitable for a linguistic solution, that is, phenomena requiring linguistic solutions, are listed below under the heading of the associated grammatical component.

(a) *Morphology:* By systematically associating each word form with its base form, all variants of a search word in the database can be found. A program of automatic word form recognition would be superior to the customary method of truncation

(b) *Lexicon:* A letter-based search does not take semantic relations between words into account. For example, the search for car would ignore relevant occurrences such as convertible, pickup truck, station wagon, and so on. A lexical structure which automatically specifies for each word the set of equivalent terms (synonyms), of the superclass (hypernyms), and of the instantiations (hyponyms) can help to overcome this weakness, especially when the domain is taken into account.

(c) *Syntax:* A letter-based search does not take syntactic structures into account. Thus, the system does not distinguish between, for example, teenagers sold used cars and teenagers were sold used cars. A possible remedy would be a syntactic parser which recognizes different grammatical relations between, for example, the subject and the object. Such a parser, which presupposes automatic word form recognition, would be superior to the currently used search for words within specified maximal distances.

(d) *Semantics:* A letter-based search does not recognize semantic relations such as negation. For example, the system would not be able to distinguish between selling cars and selling no cars. Also, equivalent descriptions of the same facts, such as A sold x to B and B bought x from A, could not be recognized. Based on a syntactic parser and a suitable lexicon, the semantic interpretation of a textual database could analyze these distinctions and relations, helping to improve recall and precision.

(e) *Pragmatics:* According to [5], a major reason for poor recall was the frequent use of context-dependent formulations such as concerning our last letter, following our recent discussion, as well as nonspecific words such as problem, situation, or occurrence. The treatment of these frequent phenomena requires a complete theoretical understanding of natural language pragmatics. For example, the system will have to be able to infer that, for example, seventeen-year-old bought battered convertible is relevant to the query used car sales to teenagers.

3.2.2 Linguistic Methods of Optimization

In order to improve recall and precision, linguistic knowledge may be applied in various different places in the database structure. The main alternatives (called linguistic methods of optimization), are whether improvements in the search should be based on preprocessing the query, refining the index, and/or post-processing the result. Further alternatives are an automatic or an interactive refinement of the query and/or the result.

4. Smart versus Solid Solutions in Computational Linguistics [1]

The different degrees of using linguistic theory for handling the retrieval from textual databases illustrate the choice between smart versus solid solutions.

4.1 Smart Solutions in Computational Linguistics

Smart solutions avoid difficult, costly, or theoretically unsolved aspects of natural communication, as in

(a) Weizenbaum's Eliza program [2], which appears to understand natural language,

but doesn't, as mentioned in the Introduction Section.

(b) direct and transfer approaches in machine translation, which avoid understanding

the source text (Sections. 5.1 and 5.2), and

(c) finite state technology and statistics for tagging and probabilistic parsing [3].

Initially, smart solutions seem cheaper and quicker, but they are costly to maintain and their accuracy cannot be substantially improved. The alternative is solid solutions:

4.2 Solid Solutions in Computational Linguistics

Solid solutions aim at a complete theoretical and practical understanding of natural language communication. Applications are based on ready-made off-the-shelf components such as

(a) online lexica,

(b) rule-based grammars for the syntactic-semantic analysis of word forms and sentences. (These methods may seem impressive because of the vast number of toys and tools assembled in the course of many decades [3]. But they do not provide an answer to the question of how natural language communication works. What is needed instead is a functional reconstruction of the engine, the transmission, the steering mechanism, and so on. That is, a solid solution.)

(c) parsers and generators for running the grammars in the analysis and production
of free text, and
(d) reference and monitor corpora for different domains, which provide a systematic,
standardized account of the current state of the language.
Solid solution components are an application-independent long-term investment. Due to their systematic theoretical structure they are easy to maintain, can be improved continuously, and may be used again and again in different applications.

5. Smart Solutions in Computational Applications [1]

5.1. Beginnings of Machine Translation

The choice between a smart and a solid solution is exemplified by machine translation. Translation in general requires understanding a text or utterance in a certain language (interpretation) and reconstructing it in another language (production).

5.1.1 Formula to Compute the Number of Language Pairs

$n \cdot (n - 1)$*, where n = number of different languages*
For example, an EU with 23 different languages has to deal with a total of $23 \cdot 22 = 506$ language pairs.
In a language pair, the source language (SL) and the target language (TL) are distinguished. For example, 'French→Danish' and 'Danish→French' are different language pairs.

5.1.2 Schema of Direct Translation and Its Weakness

Each language pair requires the programming of its own direct translation system. Direct translation is based mainly on a differentiated dictionary, distinguishing many special cases for a correct assignment of word forms in the target language.

Fully Automatic High Quality Translation (FAHQT) was just around the corner, their hopes were not fulfilled. Hutchins [6] provides the following examples to illustrate the striking shortcomings of early translation systems:

5.1.3 Example of Automatic Mistranslations

Out of sight, out of mind. ⇒*Invisible idiot.*
The spirit is willing, but the flesh is weak. ⇒ *The whiskey is all right, but the meat is rotten.*
La Cour de Justice considère la création d'un sixième poste d'avocat général.⇒*The Court of Justice is considering the creation of a sixth avocado station.*
The first two examples are apocryphal, described as the result of an automatic translation from English into Russian and back into English. The third example is documented in Lawson [11] as output of the SYSTRAN system. An attempt to avoid the weaknesses of direct translation is the transfer approach as follows:

5.2. Machine Translation Today －Interlingua Approach

The importance of language *understanding* for adequate translation is illustrated by the following examples:

5.2.1 Syntactic Ambiguity in the Source Language

1. Julia flew and crashed the airplane.
Julia (flew and crashed the airplane)
(Julia flew) and (crashed the airplane)
2. Susanne observed the yacht with a telescope.
Susanne observed the man with a beard.
3. The mixture gives off dangerous cyanide and chlorine fumes.
(dangerous cyanide) and (chlorine fumes)

dangerous (cyanide and chlorine) fumes

The first example is ambiguous between using the verb fly transitively (someone flies an airplane) or intransitively (someone/-thing flies). The second example provides a choice between an adnominal and an adverbial interpretation. The third example exhibits a scope ambiguity regarding dangerous.

5.2.2 Partial Solutions for Practical Machine Translation

1. *Machine-aided translation* (MAT) supports human translators with comfortable tools such as online dictionaries, text processing, morphological analysis, etc.
2. *Rough translation* – as provided by an automatic transfer system – arguably reduces the translators' work to correcting the automatic output.
3. *Restricted language* provides a fully automatic translation, but only for texts which fulfill canonical restrictions on lexical items and syntactic structures.

The interlingua approach is based on a general, language-independent level called the interlingua. It is designed to represent contents derived from different source languages in a uniform format. From this representation, the surfaces of different target languages are generated.

5.2.3 Schema of the Interlingua Approach

An interlingua system handles translation in two independent steps. The first step translates the source language text into the interlingua representation (analysis). The second step maps the interlingua representation into the target language (synthesis).
It follows from the basic structure of the interlingua approach that for $n(n-1)$ language pairs only $2n$ interlingual components are needed (namely n analysis and n synthesis modules), in contrast to the direct and the transfer approach which require $n(n-1)$ components. Thus, as soon as more than three languages ($n > 3$) are involved, the interlingua approach has a substantial advantage over the other two.
The crucial question, however, is the exact nature of the interlingua.

6. Experiment

6.1 Music Information Retrieval (MIR) — A Solid Solution Example at Work

Purpose: To demonstrate the functionality of the Self-Organizing-Maps (SOM)-based music content representation.

Off-the-shelf tool used: MATLAB neural networks library function musicVisualizationDemoSOM.mat.

Input data: musicLargeData dataset.

Output:

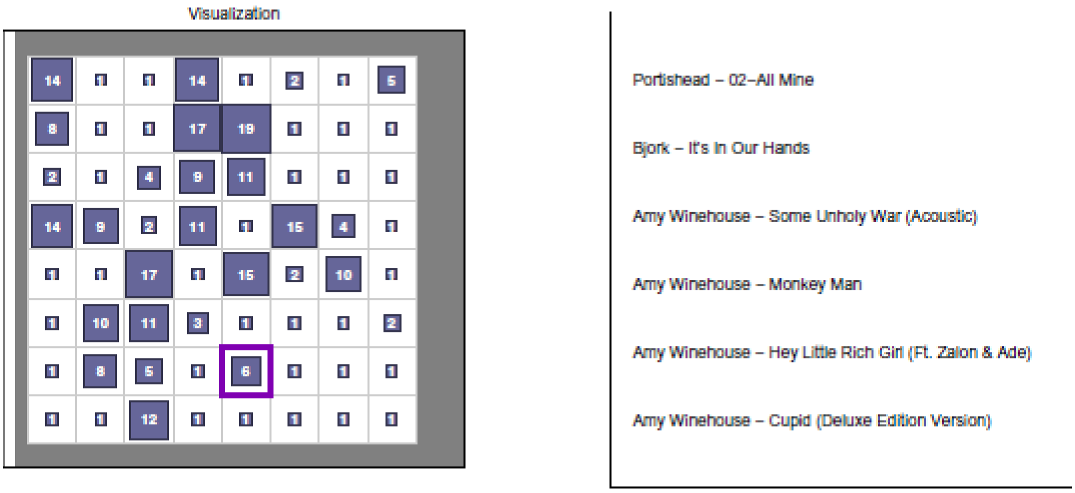
Visualization
14 1 1 14 1 2 1 5
8 1 1 17 19 1 1 1
2 1 4 9 11 1 1 1
14 9 2 11 1 15 4 1
1 1 17 1 15 2 10 1
1 10 11 3 1 1 1 2
1 8 5 1 6 1 1 1
1 1 12 1 1 1 1 1
Portishead – 02–All Mine
Bjork – It's In Our Hands
Amy Winehouse – Some Unholy War (Acoustic)
Amy Winehouse – Monkey Man
Amy Winehouse – Hey Little Rich Girl (Ft. Zalon & Ade)
Amy Winehouse – Cupid (Deluxe Edition Version)

Figure 1 Visualization of selected nodes of the SOM of the data in the musicLarge Data.mat dataset.

Figure 1 presents some node visualization examples. It can be seen that in some cases, the example contains songs with female vocalists. The rest of the examples (not shown) mostly highlight songs of the same artist. In some cases, the proximity of two nodes has a physical meaning: the 3rd and the 4th examples are (practically) homogeneous with respect to the identity of the artist, and in addition, the two artists can be considered to belong to similar musical genres (Green Day and Bad Religion both fall into the punk-rock genre).

6.2 Talk to Eliza — A Smart Solution

Purpose: Test Eliza program
Input: Dialog in writing to Eliza on a screen

Tool used: Eliza program online [8].
Output: Eliza: Tell me more. -> Me: Of course. -> Eliza: OK… Me: -> What do me think am important now. -> Eliza: We were discussing you, not me. -> Me: O.K., I missed my mother. How do you think?. -> Eliza:
Perhaps you would like to be able to tell me more. *** Are we going to chat? I cannot help you without a dialog. -> Me: What is the most important thing in life? -> Eliza: Have you asked anyone else?

6.3 Machine Translation — A Smart solution

Tool: Google online translation program [9].
Input in Chinese: 朝辭白帝彩雲間，千里江陵一日還，兩岸猿聲啼不住，輕舟已過萬重山
Output in English: In the remarks of the White Emperor, the Qianling River is still a day, and the two sides of the river are screaming. The light boat has passed the Wanzhong Mountain.
Input in English: Ask not what your country can do for you. Ask what you can do for your country.
Output in Chinese: 不要問你的國家能為你做些什麼。 問你能為國家做些什麼。
Input in Japanese: どんな公園でも、両親にとって、子供たち
Output in Chinese: 適用於任何公園，父母，兒童
Output in English: For any park, parents, children

7. Conclusions

The main reason for the long-term superiority of solid solutions, however, is quality. This is because a 75 % smart solution is typically very difficult or even impossible to improve to 76 %.。

The goal of computational linguistics, however, is a solid solution in science: it must (i) explain the mechanism of natural communication theoretically and (ii) verify the theory with an implementation (software machine) which may be loaded with the language-dependent lexicon and compositional operations of any natural language. The speak and the hear mode of the implementation must work in any practical application for which free natural language communication between humans and machines is desired [7, 8].

References

[1] Roland Hausser, *Foundations of Computational Linguistics: Human-Computer Communication in Natural Language*. 3rd ed. Springer-Verlag Berlin Heidelberg, 2014.

[2] J. Weizenbaum, "*ELIZA – A Computer Program for the Study of Natural*

LanguageCommunications Between Man and Machine,“ *Communications of the ACM* 9.1 (11), 1965.

[3] D. Jurafsky, J. Martin, “*Speech and Language Processing: An Introduction to Natural Language,”Processing, Computational Linguistics, and Speech Recognition,* Upper Saddle River: Prentice Hall, 2000.

[4] A.M. Turing, *Computing Machinery and Intelligence*, *Mind* 59:433–460, 1950.

[5] D.C. Blair, and M.E. Maron, “An Evaluation of Retrieval Effectiveness for a Full-Text Document Retrieval System,” *Communications of the ACM* 28.3:289–299, 1985.

[6] W.J. Hutchins , “*Machine Translation: Past, Present, Future*,” Chichester: Ellis Horwood, 1986.

[7] T. Giannakopoulos and A. Pikrakis, Introduction to Audio Analysis: A MATLAB Approach, Elsevier, 2014.

[8] Eliza, Computer Therapist - CyberPsych (https://www.cyberpsych.org/eliza/)

[9] https://translate.google.com.tw/

[10] https://en.wikipedia.org/wiki/Computing_Machinery_and_Intelligence

[11] V. Lawson, “Machine Translation,” in C. Picken (ed.), 203–213, 1983.