Dynamic Lexical Acquisition in Chinese Sentence Analysis

Andi Wu
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
andiwu@microsoft.com

Joseph Pentheroudakis
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
josephp@microsoft.com

Zixin Jiang
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
jiangz@microsoft.com

Abstract

Dynamic lexical acquisition is a procedure where the lexicon of an NLP system is updated automatically during sentence analysis. In our system, new words and new attributes are proposed online according to the context of each sentence, and then get accepted or rejected during syntactic analysis. The accepted lexical information is stored in an auxiliary lexicon which can be used in conjunction with the existing dictionary in subsequent processing. In this way, we are able to process sentences with an incomplete lexicon and fill in the missing info without the need of human editing. As the auxiliary lexicons are corpus-based, domain-specific dictionaries can be created automatically by combining the existing dictionary with different auxiliary lexicons. Evaluation shows that this mechanism significantly improves the coverage of our parser.

Introduction

The quality of many NLP systems depends heavily on the completeness of the dictionary they use. However, no dictionary can ever be complete since new words are being coined constantly and the properties of existing words can change over time. In addition, a dictionary can be relatively complete for a given domain but massively incomplete for a different domain. The traditional way to make a dictionary more complete is to edit the dictionary itself, either by hand or through batch updates using data obtained from other sources. This approach is undesirable because

(1) it can be very expensive due to the amount of hand work required;
(2) the job will never be complete since new words and new usages of words will continue to appear.
(3) certain words and usages of words decay after a while or only exist in a certain domain, and it is inappropriate to make them a permanent part of the dictionary.

This paper discusses an alternative approach where, instead of editing a static dictionary, we acquire lexical information dynamically during sentence analysis. This approach is currently implemented in our Chinese system and Chinese examples will be used to illustrate the process. In Section 1, we will discuss how the new lexical information is discovered. Section 2 discusses how such information is filtered, lexicalized, and used in future processing. Section 3 is devoted to evaluation.

1 Proposing words and attributes

Two major types of lexical information are being acquired dynamically in our current Chinese system: new words and new grammatical attributes such as parts of speech (POS) and sub-categorization frames. The acquisition assumes the availability of an existing dictionary which is relatively mature though incomplete in many ways. In our case, we have a lexicon of 88,000 entries with grammatical attributes in most of them. Our assumption is that, once a dictionary has reached this scale, we should have enough information to predict the missing information in the context of sentence analysis. We can then stop hand-editing the static dictionary and let dynamic lexical acquisition take over.

In most cases, the grammatical properties of a word define the syntactic context in which this word may appear. Therefore, it is often possible

to detect the grammatical properties of a word by looking at the surrounding context of this word in a sentence. In fact, this is one of the main criteria used by lexicographers, who often apply a conscious or subconscious contextual "template" for each grammatical property they assign. We have coded those templates in our system so that a computer can make similar judgments.[1] When a word is found to fit into a template for a given property but we do not have that property in the dictionary yet, we can make a guess and propose to add that property. Our current Chinese system has 29 such templates, 14 for detecting new words and 15 for detecting new grammatical attributes for new or existing words.

1.1 Proposing new words

Two types of unlisted words exist in Chinese:

(1) single-character bound morphemes used as words;

(2) new combinations of characters as words.

An example of Type (1) is 侃. This is a bound morpheme in our dictionary, appearing only as a part in words like 侃大山 (have a good chat). However, like many other bound morphemes in Chinese, it can occasionally be used as an independent word, as in the following sentence:

他 在 我 家 侃 了 两 个 小时。
he at I home chat LE two CL hour
He chatted for two hours at my house.

The usual response to this problem is to treat it as a lexical gap and edit the entry of 侃 to make it a verb in the dictionary. This is undesirable for at least two reasons. First of all, many bound morphemes in Chinese can be occasionally used as words and making all of them independent words will introduce a lot of noise in sentence analysis. Secondly, it will be a difficult task for lexicographers, not just because it takes time, but because the lexicographers will often be unable to make the decision unless they see sentences where a given bound morpheme is used as a word.

In our system, we leave the existing dictionary untouched. Instead, we "promote" a bound morpheme to be a word dynamically when it appears in certain contextual templates. The template that promotes 侃 to be a verb may include conditions such as:

- not subsumed by a longer word, such as 侃侃而谈;
- being part of an existing multiple-character verb, such as 侃 in 侃侃而谈;
- followed by an aspect marker, such as 了;
- etc.

Currently we have 4 such templates, promoting morphemes to nouns, verbs, adjectives and adverbs respectively.

Examples of Type (2) are found all the time and adding them all to the existing dictionary will be a never-ending job. Here is an example:

无 需 重新 启动 就 可以 接驳 或 解驳
not need again start then can dock or undock
便携 电脑
easy-to-carry computer
You can dock and undock your laptop without restarting.

接驳 (dock), 解驳 (undock) and 便携 (easy-to-carry) are not entries in our dictionary. Instead of adding them to the dictionary, we use templates to recognize them online. The template that combines two individual characters to form a verb may include conditions such as:

- none of the characters is subsumed by a longer word;
- the joint probability of the characters being independent words in text is low;
- the internal structure of the new word conforms to the word formation rules of Chinese
- the component characters have similar behavior in existing words
- etc.

The details can be found in Wu & Jiang (2000). Currently we have 10 such templates, which are capable of identifying nouns, verbs, adjectives and adverbs of various lengths.

[1] Currently these templates are hand-coded heuristics based on linguists' intuition. We are planning to use machine learning techniques to acquire those templates automatically.

1.2. Proposing grammatical attributes

POS and sub-categorization information is crucial for the success of sentence analysis. However, there is no guarantee that every word in the existing dictionary will have the correct POS and sub-categorization information. Besides, words can behave differently in different domains or develop new properties over time. Take the Chinese word 同步 (synchronize) for example. It is an intransitive verb in our dictionary, but it is now often used as a transitive verb, especially in the computer domain. For instance:

MADC可 方便 地 同步 Exchange 帐户。
can easily DE synchronize account
MADC (Microsoft Active Directory Connector) can easily synchronize Exchange accounts.

We may want to change the existing dictionary to make words like 同步 transitive verbs, but that may not be appropriate lexicographically, at least in the general domain, not to mention the human labor involved in such an undertaking. However, the sentence above cannot get a spanning parse unless 同步 is a transitive verb. To overcome this difficulty, our system can dynamically create a transitive verb in certain contexts. An obvious context would be "followed by an NP", for example. This way we are able to parse the sentence without changing the dictionary.

A similar approach is taken in cases where a word is used in a part of speech other than the one(s) specified in the dictionary. In the following sentence, for example, the noun 群集 (cluster) is used as a verb instead:

你 可以 群集 32 台 服务器
you can cluster 32 CL server
You can cluster 32 servers.

Rather than edit the dictionary to permanently add the verb POS to nouns like 群集, we turn them into verbs dynamically during sentence analysis if they fit into the verb template. The conditions in the verb template may include:

- preceded by an modal or auxiliary verb
- followed by aspectual markers such as 了, 过 and 着
- preceded by adverbials
- etc.

Such templates are in effect very similar to POS taggers, though we use them exclusively to create new POS instead of choosing from existing POS.

2 Harvesting new words and attributes

Proposing of new words and attributes as described in the previous section is only intended to be intelligent guesses, which can be wrong sometimes. For example, although transitive verbs tend to be followed by NPs, not all verbs that precede NPs are transitive verbs. To make sure that (1) the wrong guesses do not introduce too much noise into the analysis and (2) only the correct guesses are accepted as true lexical information, we take the following steps to filter out the errors that result from over-guessing.

2.1 Set up the competition

The proposed words and attributes are assigned lower probability in our system. This is straightforward for new words. We simply assign them low scores when we add them (as new terminal nodes) to the parsing chart[2]. For new attributes on existing words, we make a new node which is a copy of the original node and assign the new attributes and a lower probability to this node. As a result, the chart will contain two nodes for the same word, one with the new attributes and one without. The overall effect is that the newly proposed nodes will compete with other nodes to get into a parse, though with a disadvantage. The sub-trees built with the new nodes will have lower scores and will not be in the preferred analysis unless there is no other way to get a spanning parse. Therefore, if the guesses are wrong and the sentence can be successfully parsed without the additional nodes, the best parse (the parse with the highest score) will not contain those nodes and the guesses are practically ignored. On the other hand, if the guesses are right and we cannot get any successful parse unless we use them, then they will end up in the top parse[3] in spite of their low

[2] See Jensen et al (1993) and Heidorn (2000) for a general description of how chart parsing works in our system. A Chinese-specific description of the system can be found in Wu & Jiang (1998).

[3] Our system can produce more than one parse for a

probability.

2.2 Keep the winners

For each sentence, we pick the top parse and check it to see if there are any terminal nodes that are new words or nodes containing new attributes. If so, we know that these nodes are necessary at least to make the current sentence analyzable. The fact that they are able to beat their competitors despite their disadvantage suggests that they probably represent lexical information that is missing in the existing dictionary. We therefore collect such information and store it away in a separate lexicon. This auxiliary lexicon contains entries for the new words and the new attributes of existing words. Each entry in this lexicon carries a frequency count which records the number of times a given new word or new attribute has appeared in good parses during the processing of certain texts. The content of this lexicon depends on the corpora, of course, and different lexicons can be built for different domains. When processing future sentences, the entries in those lexicons can be dynamically merged with the entries in the main lexicon, so that we do not have to make the same guesses again.

2.3 Use the fittest

The information lexicalized in those auxiliary lexicons, though good in general, is not guaranteed to be correct. While being necessary for a successful parse is strong evidence for its validity, that is not a sufficient condition for the correctness of such information. Consequently, there can be some noise in those lexicons. However, a real linguistic property is likely to be found consistently whereas mistakes tend to be random. To prevent the use of wrongly lexicalized entries, we may require a frequency threshold during the merging process: only those entries that have been encountered more than *n* times in the corpora are allowed to be merged with the main lexicon and used in future analysis. If a given new word or linguistic property is found to occur repeatedly across different domains, we may even consider physically merging it into the main dictionary, as it may be a piece of information that is worth adding permanently.

given sentence and the top parse is the one with the highest score.

3 Evaluation

The system described above has been evaluated in terms of the contribution it makes in parsing. The corpus parsed in the evaluation consists of 121,863 sentences from Microsoft technical manuals. The choice is based on the consideration that this is a typical domain-specific text where there are many unlisted words and many novel usages of words.[4] To tease apart the effects of online guessing and lexicalization, we did two separate tests, one with online guessing only and one with lexicalization as well. When lexicalization is switched on, the new words and attributes that are stored in the auxiliary lexicon are used in subsequent processing. Once a new word or attribute has been recognized in *n* sentences, it will act as if it were an entry in the main dictionary and can be used in the analysis of any other sentence with normal probability.

3.1 Online guessing only

In this test, we parsed the corpus twice, once with guessing and once without. Then we picked out all the sentences that had different analyses in the two passes and compared their parses to see if they became better when lexical guessing is on. Since comparing the parses requires human inspection and is therefore very time consuming, we randomly selected 10,000 sentences out of the 121,863 and used only those sentences in the test.

It turns out that 1,459 of those 10,000 sentences got different parses when lexical guessing is switched on. Human comparison of those differences shows that, of the 1,459, the guessing made 1,153 better, 82 worse, and 224 stay the same (different parses but equally good or bad). The net gain is 1,071. In other words, 10.71% of the sentences became better when lexical guessing is used.

[4] The novel usages are mainly due to the fact that the text is translated from English.

More detailed analysis shows that 48% of the improvements are due to the recognition of new words and 52% to the addition of new grammatical attributes. Of the 82 sentences that became worse, 6 failed because of the lack of storage during processing caused by the additional resources required by the guessing algorithm. The rest are due to over-guessing, or more precisely, the failure to rule out the over-guesses in sentence analysis. The guessing component is designed to over-guess, since the goal there is recall rather than precision. The latter is achieved by the filtering effect of the parser.

3.2 Additional gain with lexicalization

In this second test, we evaluated the effect of lexicalization on new word recognition[5]. We parsed all the 121,863 sentences twice, once with lexicalization and once without. The number of unique new words recognized in this corpus is 922[6]. Notice that this number does not change between the two processes. Using the lexicon created by dynamic lexicalization will increase the instances of those words being recognized, but will not change the number of unique words, since the entries in the auxiliary lexicon can also be recognized online. However, the numbers of instances are different in the two cases. When lexicalization is turned off, we are able to get 5963 instances of those 922 new words in 5239 sentences. When lexicalization is on, however, we are able to get 6464 instances in 5608 sentences. In other words, we can increase the recognition rate by 8.4% and potentially save 369 additional sentences in parsing. The reason for this improvement is that, without lexicalization, we may fail to identify the new words in certain sentences because there were not enough good contexts in those sentences for the identification. Once those words are lexicalized, we no longer have to depend on context-based guessing and those sentences can benefit from what we have learned from other sentences. Here is a concrete example for illustration:

他 掌握 了 解驳 便携电脑 的 技术。
He master LE undock laptop DE technology
He mastered the technology of undocking a laptop.

In this sentence, we do not have enough context to identify the new word 解驳 because 了解 is a word in Chinese (Remember there are no spaces between words in Chinese!). This destroys the condition that none of the characters in the new word should be subsumed by a longer word. However, if 解驳 has been recognized in some other sentences, such as the one we saw in Section 1.1, and has been lexicalized, we can simply look up this word in the dictionary and use it right away. In short, lexicalization enables what is learned locally to be available globally.

Conclusion

In this paper, we have demonstrated a mechanism for dynamic dictionary update. This method reduces human effort in dictionary maintenance and facilitates domain-switching in sentence analysis. Evaluation shows that this mechanism makes a significant contribution to parsing, especially the parsing of large, domain-specific corpora.

References

Heidorn G. E. (2000) *Intelligent writing assistance*,. In "A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text ", Dale R., Moisl H., and Somers H. eds., Marcel Dekker, New York, pp. 181-207.

Jenson K., Heidorn G. and Richardson S. (1993) *Natural Language Processing: the PLNLP Approach" Boston, Kluwer*

Wu A. and Jiang Z. (1998) *Word Segmentation in Sentence Analysis.* In "Proceedings of the 1998 International Conference on Chinese Information Processing", Beijing, China.

Wu A. and Jiang Z. (2000) *Statistically-Enhanced New Word Identification in a Rule-based Chinese System.* In "Proceedings of the Second ACL Chinese Processing Workshop", HKUST, Hong Kong.

[5] We would like to look at the effect on grammatical attributes as well, but the evaluation is not as straightforward there and much more time-consuming.

[6] The total number of unique words used in this corpus is 17,110. So at least 5% of the words are missing in the original dictionary.