Collaborative Annotation and Visualization of Functional and Discourse Structures

Hengbin Yan
Halliday Centre for Intelligent Applications of Language Studies,
Department of Chinese, Translation and Linguistics,
City University of Hong Kong
hbyan2@cityu.edu.hk

Jonathan Webster
Halliday Centre for Intelligent Applications of Language Studies,
Department of Chinese, Translation and Linguistics,
City University of Hong Kong
ctjjw@cityu.edu.hk

Abstract

Linguistic annotation is the process of adding additional notations to raw linguistic data for descriptive or analytical purposes. In the tagging of complex Chinese and multilingual linguistic data with a sophisticated linguistic framework, immediate visualization of the complex multi-layered functional and discourse structures is crucial for both speeding up the tagging process and reducing errors. The need for large-scale linguistically annotated corpora has made collaborative annotation increasingly essential, and existing annotation tools are inadequate to the task of providing assistance to annotators when dealing with complex linguistic structural information. In this paper we describe the design and development of a collaborative tool to extend existing annotation tools. The tool improves annotation efficiency and addresses certain difficulties in representing complex linguistic relations. Here, we adopt annotation based on Systemic Functional Linguistics and Rhetorical Structure Theory to demonstrate the effectiveness of the interface built on such infrastructure.

Keywords: Linguistic Annotation, Linguistic Visualization, Cross-domain References

1. Introduction

Recent years have witnessed an increasing need for large-scale high-quality annotated corpora on complex Chinese linguistic information where no automated annotators are available. Annotation on multi-level data complex structural relationships in such linguistic frameworks as Systemic Functional Grammar (SFG) [1] and Rhetorical Structure Theory [2] is a difficult task.

SFG investigates texts as intentional acts of meaning, organized in functional-semantic components known as “metafunctions”. Three primary metafunctions, operating in parallel and each representing a layer of meaning with a set of options to language users, cover different functional aspects of human communication and expression: the *ideational*,

interpersonal and *textual* metafunctions. For our purposes our discussion will focus on analysis and annotation of these three metafunctions in SFG.

Despite the fact that SFG is becoming increasingly influential among Chinese linguistic researchers, a large-scale, high-quality corpus annotated with SFG has yet to be developed [3]. Consequently, when trying to conduct corpus-based analysis using the SFG framework researchers must either 1) spend an enormous amount of time studying an unannotated corpus, 2) embark on the error-prone process of manually annotating a corpus on their own, or 3) rely on small corpora independently annotated by researchers which may not be particularly suited to needs of the tasks at hand.

The lack of high-quality Chinese SFG corpora is partly attributable to the lack of a competent SFG tagger capable of annotating large-scale corpora while ensuring quality. In developing such a tagger, a number of challenges need to be addressed:

1) Lack of an efficient and sophisticated storage scheme for storing such multilayered information with complex structures
2) Additional visual cues to facilitate the tagging process
3) Need for collaborative tasking (co-tasking) by different annotators

The most common method to annotate text includes the use of an open standard like XML document. Provided one possesses the prerequisite familiarity with XML conventions, the linguist-as-annotator inserts metadata most likely using a plain-text editor or generic XML editor. This method works well so long as the text is short, and the required linguistic information is relatively simple. While some special editing tools have been created which provide a graphical interface for linguists to tag texts, such tools, for the most part, tend to be stand-alone, primarily oriented to single users.

To facilitate efficient, high quality annotation of a large amount of Chinese text material by a team of co-tasking linguists, we have developed a new multi-user linguistic information annotator, which provides real-time cross-domain reference as visual “feedback”, thereby assisting linguists to tag text data in a highly effective way. Multiple users can work at the same time on any portion of the text, with their annotations revealed (or selectively not) to other members as reference. Those responsible for verification, comparison, correction, and progress tracking can view the work even as it is being carried out. This design is intended to improve both the efficiency and quality of annotation, while enabling multi-user tagging of substantially greater text material in shorter time.

2. The Framework

Here, we first review existing tools for annotating texts before discussing the advantages of our new tool. We also present an application scenario of our tools for annotating text and explain how visualized cross-domain reference works.

A number of similar tools have been developed for various annotation scenarios. MMAX2 [4] is a customizable tool for creating and visualizing multilevel linguistic annotations that allows outputs the results of annotations according to predefined style-sheets. It supports tagging of part-of-speech tags, coreference and grammatical relations, but is not capable of representing and visualizing complex discourse level structures. SALTO [5] is a multilevel annotation tool for annotating semantic roles and Treebank syntactic structures. O’Donnell’s annotation tool for Systemic Functional Linguistics, the UAM CorpusTool [6], is intended for annotating multi-layered Systemic Functional Grammar structures by a Single User. Both tools are restrictive in terms of functionalities and do not support collaborative annotation and provide no means of representing complex sentential structures.

Our representation model is built on the functionalities of Annotation Graph [7] and the underlying storage scheme is conceptually similar to Standoff XML format [9], but we opted

for a relational database structure built with an object-oriented design for efficiency, reusability and versatility.
Several web-based annotation tools such as Serengeti [10], a tool for annotating anaphoric relations and lexical chains, are limited to a particular domain and cannot be used for annotating and visualizing complex structural information without substantial modification.

2.1 Web-based Collaborative Annotation

Traditionally, annotation processes that involve more than one annotator are often divided into multiple steps where one step is taken up and completed by one annotator before being passed on to another. This is adequate for small annotation projects where only a linear sequential procedure is involved. In recent years, however, the growing scale and complexity of annotation projects have necessitated the collaboration of different annotators who are often geographically dispersed. In view of these needs, we develop our application on a web-based infrastructure making it accessible from any web-accessible point and enabling collaborative annotation on the same data source either synchronously or asynchronously.
One problem that arises in collaborative annotation is that annotators often come with different sets of skills and have varying, sometimes overlapping responsibilities. Our goal is provide a user-friendly, intuitive interface, designed to reduce the drudgery of XML-based annotation, while enforcing annotating standards and quality functionalities for user management and versioning.
Each stage in the annotation process is divided into several hierarchically structured steps in which each parent step can spawn child steps to be taken up by one or more annotators. This gives the annotator fine-grained control over the annotating process and facilitates clear division of labor among different annotators. In addition, all annotators collaborating on the same step get notified of the relevant changes in annotation in real time once a modification has been made.
The tagger is built on a generic, multifunctional relational database similar to the annotation graph model [7] that has been demonstrated to be capable of representing virtually all sorts of common linguistic annotations. In the collaborative environment annotators can plug in certain linguistic resource that can serve as the standardized version assessable to all annotators, instead of each annotator keeping his own version, which may cause severe merging difficulties.

2.2 Representation of Complex Linguistic Structures and Relationships

The storage scheme for traditional annotation tools built using XML have been largely restrained by the inherent limitations of XML, which is suitable for storing written texts that are continuous, linear and single-layered. For non-continuous, overlapping and multi-layered linguistic information, XML-based tools typically rely on complex workarounds that unnecessarily overcomplicate the data model.
Most linguistic structures can be represented with an Annotation Graph interface. In annotating corpora with linguistic models such as Systemic Functional Grammar, where the linguistic information is structured in a multi-layered, overlapping hierarchy with references pointing to the linguistic elements, the underlying representation model must be carefully designed. The underlying data model of our platform is built on the same principles as Annotation Graph but adopts a modularized design to cover emerging use cases.
In annotating any sizable corpora, one recurring problem is representing the complex relations across various layers of linguistic elements. In this paper we have generalized common linguistic relations on three levels of linguistic elements, namely:
1) Unit Level: single linguistic elements (word, morpheme)

2) Segment Level: continuous range of linguistic elements (phrases, clauses, sentences, and paragraphs)
3) Group Level: groups of ranges of linguistic Elements (non-continuous grammatical units, i.e., clausal relations, hierarchical discourse trees in RST)

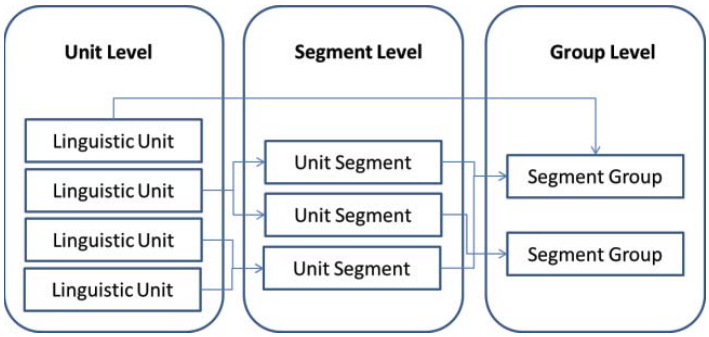

Figure 1: Three primary levels of linguistic relations.

Figure 1 illustrates a simplified abstract view of the three-level structure. At the Unit Level, the basic linguistic elements (e.g. words, morphemes) are either broken up into several separate linguistic segments, or joined together by an unlimited number of continuous units into a common segment. For example, the word *uncovered* can be made up of several morphemes (i.e., un + cover + ed), each represented by a single segment, or it can be joined together by another word (e.g. cases) to form a new segment (uncovered cases). At the Segment Level, segments (e.g. morphemes, words) can be part of a larger segment (e.g. clauses, paragraphs) in an indefinitely recursive and hierarchical manner. The Group Level is a generic structure that deals with relations among linguistic units and segments. For example, in RST there are different discourse relations (e.g. Antithesis, Condition) and roles (e.g. Nucleus, Satellite). Such relations in the data model are defined as groups, with one textual segment pointing to another and attaching a relation (function, tag, or role) to the pointed segment. Similar to segments, the number of segments in each group is unlimited and the group as a whole can in turn be pointed to by another group with an arbitrary depth of recursion and hierarchy, but unlike units in segments, the segments in each group can be non-continuous and overlapping, thus enabling any complex relations to be aptly defined.
In our application scenarios, we focus on annotating hierarchical discourse structures in RST and the three layers of metafunctions in SFG. These layers of linguistic units and the complex relations among them are represented using the proposed common structure.
In one-to-many and many-to-many relations, a sequence of ordered linguistic objects may be linked across different layers. Such interrelationships can form complex linguistic networks representing intricate linguistic meanings. Due to their inherent complexity, understanding such relationships can pose challenges to annotators, especially when such relationships are constantly added or removed in a collaborative annotating environment. The platform introduces real-time visualization of the structural relations as the annotation progresses, allowing the annotator to keep track of and make changes to annotations accordingly.
In annotating such structural relations, each unit is given a unique identifying number which we use for easy grouping of the units and to define the complex, often embedded interrelations between the units (e.g., in SFG these include logico-semantic relations such as *Parataxis* and *Hypotaxis*, *Elaboration* and *Extension* etc.).

2.3 Visualized Cross-domain Reference

While the past decade has seen significant advancement in the automatic annotation of

functional structures, the automatic annotation of semantic and discourse information has been largely ignored. One difficulty has been the lack of high-quality corpora to bootstrap the automation, a time and cost extensive task that has to be done manually. In a collaborative environment, leveraging the resources of non-expert annotators can significantly boost the annotation efficiency, as has been demonstrated by recent experiments [11]. The lack of sufficient linguistic expertise, however, restrains non-expert linguistic annotators from engaging in more complex annotations. The annotation process can be significantly accelerated using assistance and reference tools such as a tag dictionary [12]. Different annotators may form different opinions on particular annotations based on their own reference to acquired linguistic knowledge. By unifying the source of such knowledge, we may be able to boost inter-annotator agreement on issues where they otherwise differ. Our annotation tool is built on a generic infrastructure compatible with various formats of linguistic information such as Treebanks, multilingual corpora, part-of-speech (POS) annotation and output from statistical syntactic parsers such as the Stanford parser. These additional corpora and annotations not only serve to enrich textual data with additional layers of linguistic information but can be potentially used to assist in annotation. In our current application scenarios, when annotating a corpus the annotator is often faced with the following tasks:

1) Divide the text into meaningful segments
2) Analyze the segmented texts for the internal structure, such as functional structure of a clause or sentence
3) Analyze the functions of each functional/semantic unit, such as the part-of-speech of each word
4) Refer to a previously annotated section similar to the one being annotated
5) Consult a thesaurus for entries to the words whose meaning is unclear
6) Consult a multilingual corpus parallel to or aligned with the corpus (when annotating a corpus in another language).

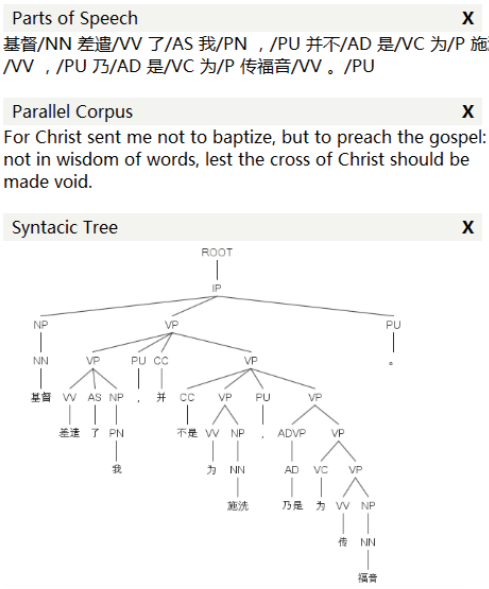

Figure 2: Automatically generated Reference Channels for annotation.

Figure 2 is an example of some of the available information that has been incorporated into our annotation platform to provide easy access for collaborating annotators. The panel is made up of three selected components that assist in the annotation task. The first section is

produced from an automatic part-of-speech (POS) tagger (we use the Stanford POS tagger). The tagger reads raw text as input and yields the POS tags of each word. This information is useful as it provide basic disambiguation and guidance when annotating the text. Similarly, the third section is produced by a syntactic parser (Stanford Parser), which not only parses the text syntactically, but generates the complete tree structure of the parse. Glancing at the tree can provide helpful information in understanding the text at a syntactic level. Both the tagger and parser are highly generic and customizable. They can be used for tagging and parsing different languages after being trained on data of corresponding languages. The second section, on the other hand, is specific to texts with corresponding translations. The example is taken from a text from the Bible, which comes with many different versions that were aligned to each other using a special mechanism.

With such information integrated with the database, it needs to be easily accessible to aid in annotation and revision (correcting errors made in the annotation). Visualization has been found to be effective in helping users process new information [13] so introducing visualization techniques to our platform should enable users to more effectively process such information. Each of the above-mentioned layers of extra information is visualized in a windowed interface that can be customized for the needs of a particular task. The annotator can decide which of the available layers to use for reference, and at different stages of annotation different layers may be presented. The visualization is an automatic process requiring no manual intervention apart from initial settings. When the annotation moves on to the next section/stage, the contents of the visualization will be automatically updated.

When designing the annotation platform we have several goals in mind: it must be intuitive and easy-to-use. The learning curve must be kept to a minimum. We reduce the process of annotation to a two-step process: 1) define the annotation range 2) assign a label. We allow optional features such as defining the step hierarchy, placing labels in each step, visualizing and editing existing annotations, defining complex linguistic relations.

In addition, it must provide immediate feedback through visualization. In functional grammar systems such as Systemic Function Grammar when tagging a particular layer of meaning, the other layers as defined in the step hierarchy should be immediately visible in a multilayered structured format. These information layers provide additional references to the current layer being annotated, especially when they are closely linked in terms of function or meaning. When errors are made they are visible from the reference panel and appropriate actions such as deletion or modification can be taken. Figure 3 shows the annotation interface we designed to meet these requirements.

Figure 3 is an illustration of some of the functionalities currently implemented. The annotator starts by selecting a range of text to annotate. Visual channels appear to assist annotators in making the decisions more easily and with a higher degree of consistency. The channels on the right side of the interface provide a detailed collection of functional and semantic labels. The label structure for a particular annotation is shown at the bottom right where the structure of different metafunctions of the selected annotation is shown in a uniform way. The annotator can operate on the labeled structure directly by adding, removing and modifying the labels in the visual structure.

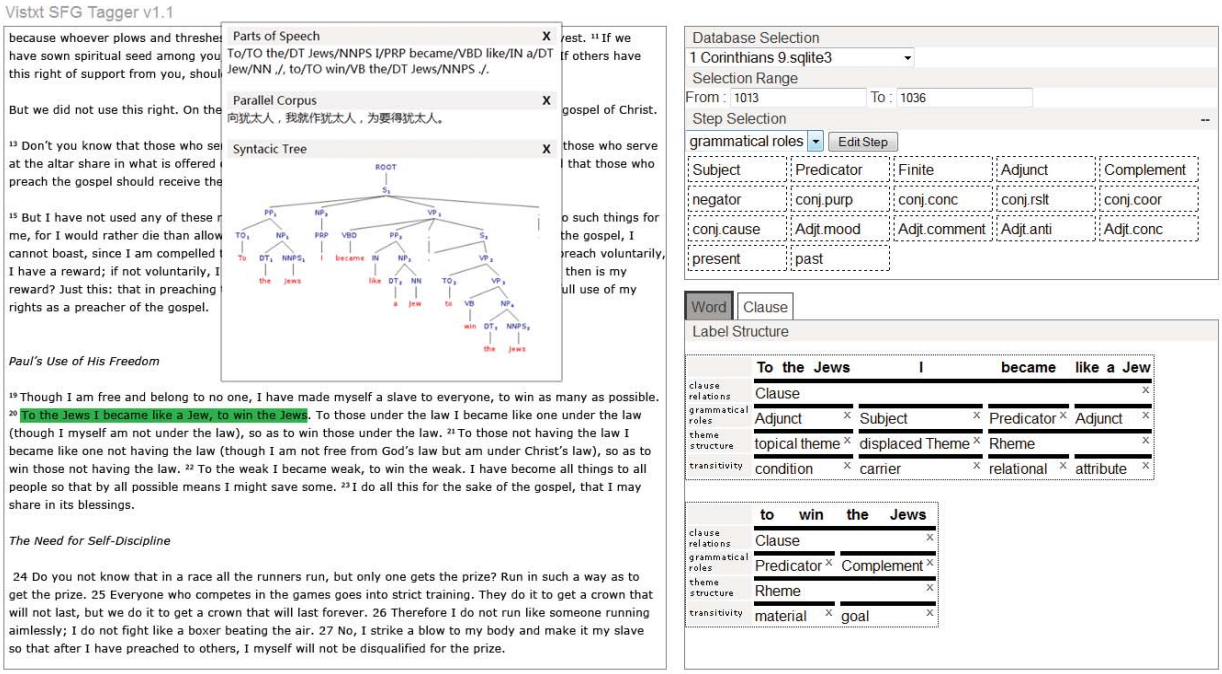

Figure 3: The web-based annotation interface.

3. APPLICATION

The tagger built on the proposed infrastructure can be used for visualizing various types of analysis. Rhetorical Structure Theory, for example, has been adopted for the tagger to help visualize the analysis of US President Obama's speeches.

Rhetorical Structure Theory (RST) is "an abstract set of convention" which "provides a general way to describe the relations among clauses in a text" [2]. This theory is widely used for text analysis for complex multilayer sentence and paragraph relations.

These sentence/paragraph relations are tagged using the proposed tagger, visualized and presented with the help of the "RST generator" which generates the RST figures, visualizing sentence/paragraph interpretation pictures.

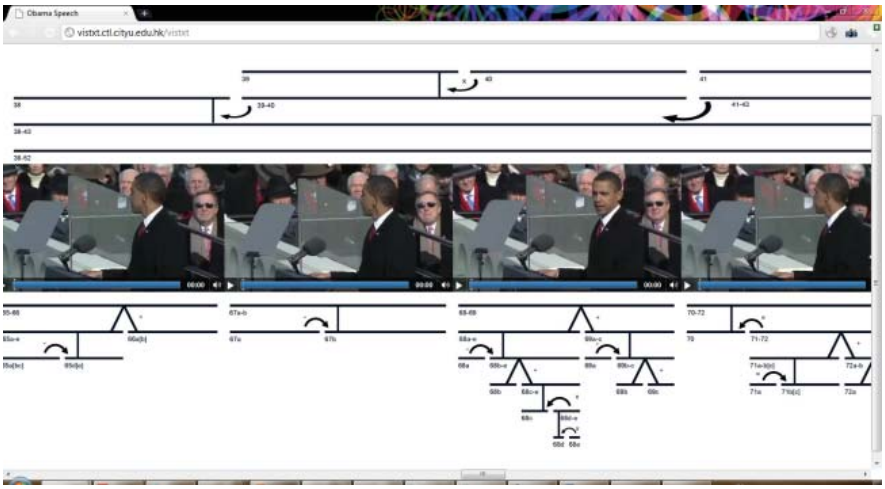

Figure 4: Visualized textual structures based on RST tagger outputs.

This annotating and visualizing method has already been applied in the analysis of Obama's inaugural and victory speeches, rendering 'the big picture' for how these speeches were

constructed (Figure 4).

4. CONCLUSION

In this paper, we present a collaborative tool for Chinese and multilingual linguistic structure annotation with visualized cross-domain references. We begin by a discussion on current trends in annotating corpora and the requirements for developing a new annotation tool. A review of existing linguistics analysis tool is presented in our introduction.

We demonstrate with example applications that 1) a large collaborative annotation platform is necessary for speeding up large-scale manual or semi-automated Chinese linguistic annotation; 2) annotating complex linguistic information is a difficult and error-prone process; 3) visualized annotation references for language structures can help facilitate the annotation process, especially in a collaborative environment; and 4) cross-domain references can further assist annotators in making the right decisions.

Our tool is designed with collaborative tasking and cross-domain analysis in mind. All linguistic signals are converted into interoperable database structures in real time when users submit their input. Data obtained from different domains can be stored in the database structure and used to serve as the basis for cross-domain references. The use of our tools for handling these relationships requires a minimal learning curve. The same system may also be used for educational purposes like annotation training and examination marking for students. Usage examples may include exercises on identifying SFL constituents, translation alignment and other language analysis.

REFERENCES

[1] M. A. Halliday and C. M. Matthiessen, “An introduction to functional grammar,” London: Edward Arnold, 2004.

[2] W. C. Mann and S. A. Thompson, “Rhetorical structure theory: Toward a functional theory of text organization,” Text, vol. 8, no. 3, pp. 243–281, 1988.

[3] M. Honnibal and J. R. Curran, “Creating a systemic functional grammar corpus from the Penn treebank,” Proceedings of the Workshop on Deep Linguistic Processing - DeepLP ’07, no. June 2005, p. 89, 2007.

[4] C. Müller and M. Strube, “Multi-level annotation of linguistic data with MMAX2,” Corpus Technology and Language Pedagogy: New, pp. 197–214, 2006.

[5] A. Burchardt, K. Erk, A. Frank, A. Kowalski, S. Pado, and M. Pinkal, “SALTO–a versatile multi-level annotation tool,” in Proceedings of LREC 2006, 2006, pp. 517–520.

[6] M. O’Donnell, “Demonstration of the UAM CorpusTool for text and image annotation,” Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies Demo Session - HLT ’08, no. June, pp. 13–16, 2008.

[7] X. Ma, H. Lee, S. Bird, and K. Maeda, “Models and tools for collaborative annotation,” Arxiv preprint cs/0204004, 2002.

[8] S. Dipper, “XML-based Stand-off Representation and Exploitation of Multi-Level Linguistic Annotation,” German Research, no. 03.

[9] S. Dipper and G. Michael, “Accessing Heterogeneous Linguistic Data — Generic

XML-based Representation and Flexible Visualization," Computational Linguistics, 2004.

[10] M. Stührenberg, D. Goecke, N. Diewald, A. Mehler, and I. Cramer, "Web-based annotation of anaphoric relations and lexical chains," in Proceedings of the Linguistic Annotation Workshop on - LAW '07, 2007, no. June, pp. 140–147.

[11] J. Chamberlain, U. Kruschwitz, and M. Poesio, "Constructing an anaphorically annotated corpus with non-experts," Proceedings of the 2009 Workshop on The People's Web Meets NLP Collaboratively Constructed Semantic Resources - People's Web '09. Association for Computational Linguistics, Morristown, NJ, USA, pp. 57–62, 2009.

[12] M. Carmen, P. Felt, R. Haertel, D. Lonsdale, O. Merkling, E. Ringger, and K. Seppi, "Tag Dictionaries Accelerate Manual Annotation," Interface, pp. 1660–1664, 2006.

[13] C. Collins, "Interactive Visualizations of natural language," PhD thesis, 2010.