A Concrete Chinese NLP Pipeline

Nanyun Peng, Francis Ferraro, Mo Yu, Nicholas Andrews,
Jay DeYoung, Max Thomas, Matthew R. Gormley, Travis Wolfe,
Craig Harman, Benjamin Van Durme, Mark Dredze
Human Language Technology Center of Excellence
Johns Hopkins University, Baltimore, Maryland USA

Abstract

Natural language processing research increasingly relies on the output of a variety of syntactic and semantic analytics. Yet integrating output from multiple analytics into a single framework can be time consuming and slow research progress. We present a CONCRETE Chinese NLP Pipeline: an NLP stack built using a series of open source systems integrated based on the CONCRETE data schema. Our pipeline includes data ingest, word segmentation, part of speech tagging, parsing, named entity recognition, relation extraction and cross document coreference resolution. Additionally, we integrate a tool for visualizing these annotations as well as allowing for the manual annotation of new data. We release our pipeline to the research community to facilitate work on Chinese language tasks that require rich linguistic annotations.

1 Introduction

Over the past few years, the natural language processing community has shifted its attention towards the Chinese language, with numerous papers covering a range of NLP tasks for Chinese. Last year's EMNLP and ACL alone featured two dozen papers focused primarily on Chinese data[1], not including many others that considered Chinese language data within a broader context. The large number of Chinese speakers, coupled with the unique challenges of Chinese compared to well studied Romance and Germanic languages, have driven these research efforts. This focus has given rise to new NLP systems that enable the automated processing of Chinese data. While some pipelines cover multiple tasks, such as Stanford CoreNLP (Manning et al., 2014), other tasks such as relation extraction are not included.

Modern NLP research, including research focused on Chinese, often relies on automatically produced analytics, or annotations, from multiple stages of linguistic analysis. Downstream systems, such as sentiment analysis and question answering, assume that data has been pre-processed by a variety of syntactic and semantic analytics. Consider the task of knowledge base population (KBP), in which information is extracted from text corpora for inclusion in a knowledge base. Associated information extraction systems rely on various NLP analytics run on the data of interest, such as relation extractors that require the identification of named entities and syntactically parsed text. Similarly, entity linking typically assumes the presence of within document coreference resolution, named entity identification and relation extraction. These analytics themselves rely on other core NLP systems, such as part of speech tagging and syntactic parsing.

While each of these tasks have received extensive attention and have associated research software for producing annotations, the output of these components must be integrated into a single cohesive framework for use in a downstream task. This integration faces a wide variety of challenges resulting from the simple fact that most research systems are designed to produce good performance on an eval-

[1] Excluding the Chinese Restaurant Process.

uation metric, but are not designed for integration in a pipeline. Beyond the production of integrated NLP pipelines, research groups often produce resources of corpora annotated by multiple systems, such as the Annotated Gigaword Corpus (Napoles et al., 2012). Effective sharing of these corpora requires a common standard.

These factors lead to the recent development of CONCRETE, a data schema that represents numerous types of linguistic annotations produced by a variety of NLP systems (Ferraro et al., 2014). CONCRETE enables interoperability between NLP systems, facilitates the development of large scale research systems, and aids sharing of richly annotated corpora.

This paper describes a Chinese NLP pipeline that ingests Chinese text to produce richly annotated data. The pipeline relies on existing Chinese NLP systems that encompass a variety of syntactic and semantic tasks. Our pipeline is built on the CONCRETE data schema to produce output in a structured, coherent and shareable format. To be clear, our goal is *not* the development of new methods or research systems. Rather, our focus is the integration of multiple tools into a single pipeline. The advantages of this newly integrated pipeline lie in the fact that the components of the pipeline communicate through a unified data schema: CONCRETE. By doing so, we can

- easily switch each component of the pipeline to any state-of-the-art model;
- keep several annotations of the same type generated by different tools; and
- easily share the annotated corpora.

Furthermore, we integrate a visualization tool for viewing and editing the annotated corpora. We posit all the above benefits as the contributions of this paper and hope the efforts can facilitate ongoing Chinese focused research and aid in the construction and distribution of annotated corpora. Our code is available at `http://hltcoe.github.io`.

2 The CONCRETE Data Schema

We use CONCRETE, a recently introduced data schema designed to capture and layer many different types of NLP output (Ferraro et al., 2014).[2] A primary purpose of CONCRETE is to ease analytic pipelining. Based on Apache Thrift (Slee et al., 2007), it captures NLP output via a number of interworking structs, which are translated automatically into in-memory representations for many common programming languages, including Java, C++ and Python. In addition to being, in practice, language-agnostic, CONCRETE and Thrift try to limit programmer error: Thrift generates I/O libraries, making it easy for analytics to read and write CONCRETE files; with this common format and I/O libraries, developers can more easily share NLP output. Unlike XML or JSON, Thrift's automatic validation of strongly typed annotations help ensure legitimate annotations: developers cannot accidentally populate a field with the wrong type of object, nor must they manually cast values.

CONCRETE allows both within-document and cross-document annotations. The former includes standard tagging tasks (e.g., NER or POS), syntactic parses, relation extraction and entity coreference, though Ferraro et al. (2014) show how CONCRETE can capture deeper semantics, such as frame semantic parses and semantic roles. These within-document annotations, such as entity coref, can form the basis of cross-document annotations.

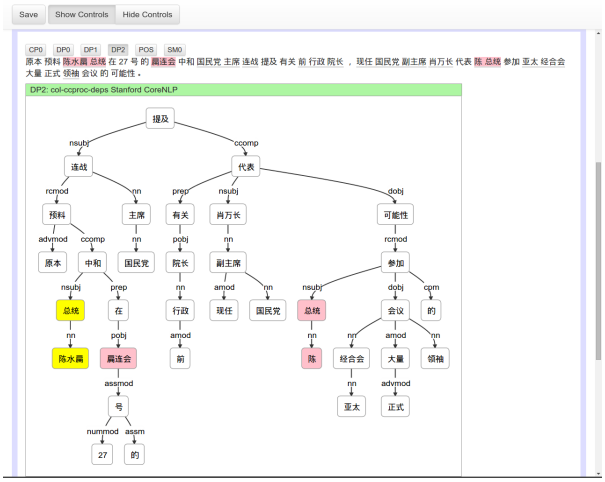
We chose CONCRETE as our data schema to support as many NLP analytics as possible. In the future, we plan to add additional analytics to our pipeline, and we expect other research groups to integrate their own tools. A flexible and well documented data schema is critical for these goals. Furthermore, the release of multiple corpora in CONCRETE (Ferraro et al., 2014) support our goal of facilitating the construction and distribution of new Chinese corpora.

3 Analytic Pipeline

We describe each stage of our pipeline with a brief description of the associated tool and relevant details of its integration into the pipeline.

[2] CONCRETE, language interfaces, and utility libraries are open-source projects (`https://hltcoe.github.io/`).

(a) The basic visualization of a `Communication`. Each line is a tokenized sentence, with options to view the part of speech, constituency and dependency parse, and entity relation information.



(b) Multiple types of annotations can be viewed simultaneously. Here, entity information is laid atop a dependency parse. A particular mention-of-interest is shown in yellow, with all other mentions in pink.

Figure 1: CONCRETE `Communication` containing Chinese text displayed in Quicklime (section 3.7).

3.1 Data Ingest

The first stage of our pipeline requires ingesting existing Chinese text into CONCRETE `Communication` objects, the core document representation of CONCRETE. The existing CONCRETE Java and Python utilities support ingesting raw text files. Part of this process requires not only ingesting the raw text, but identifying section (paragraph) and sentence boundaries.

Not all corpora contain raw text, as many corpora come with existing manual (or automatic) linguistic annotations. We provide code to support two data formats of existing Chinese corpora: the Chinese ACE 2005 relation extraction dataset (Walker et al., 2006) and the new Chinese Entities, Relations, and Events (ERE) dataset (Consortium, 2013). Both data sets include annotations for entities and a variety of relations (Aguilar et al., 2014). The labeled entities and relations are represented by CONCRETE `EntityMentions` and stored in a `EntityMentionSetList`. Additional annotations that are typically utilized by relation extraction systems, such as syntactic parses, are provided automatically by the pipeline.

3.2 Word Segmentation

Chinese text processing requires the identification of word boundaries, which are not indicated in written Chinese as they are in most other languages. Our word segmentation is provided by the Stanford CoreNLP[3] (Manning et al., 2014) Chinese word segmentation tool, which is a conditional random field (CRF) model with character based features and lexicon features according to Chang et al. (2008). Word segmentations decisions are represented by CONCRETE `Token` objects and stored in the `TokenList`. We follow the Chinese Penn Treebank segmentation standard (Xue et al., 2005). Our system tracks token offsets so that segmentation is robust to unexpected spaces or line breaks within a Chinese word.

3.3 Syntax

Part of speech tagging and syntactic parsing are also provided by Stanford CoreNLP. The part of speech tagger is based on Toutanova et al. (2003) adapted for Chinese, which is a log-linear model underneath. Integration with CONCRETE was facilitated by the concrete-stanford library [4], though supporting Chinese required significant modifications to the

[3] `http://nlp.stanford.edu/software/corenlp.shtml`
[4] `https://github.com/hltcoe/concrete-stanford`

library. Resulting tags are stored in a CONCRETE `TokenTaggingList`.

Syntactic constituency parsing is based on the model of Klein and Manning (2003) adapted for Chinese. We obtained dependency parses from the CoreNLP dependency converter. We store the constituency parses as a CONCRETE `Parse`, and the dependency analyses as CONCRETE `DependencyParses`.

3.4 Named Entity Recognition

We support the two most common named entity annotation standards: the CoNLL standard (four types: person, organization, location and miscellaneous), and the ACE standard, which includes the additional types of geo-political entity, facility, weapon and vehicle. The ACE standard also includes support for nested entities. We used the Stanford CoreNLP NER toolkit which is a CRF model based on the method in Finkel et al. (2005), plus features based on Brown clustering. For the CoNLL standard annotations, we use one CRF model to label all the four types of entities. For the ACE standard annotations, in order to deal with the nested cases, we build one tagger for each entity type. Each entity is stored in a CONCRETE `EntityMention`.

3.5 Relation Extraction

Relations are extracted for every pair of entity mentions. We use a log-linear model with both traditional hand-crafted features and word embedding features. The hand-crafted features include all the baseline features of Zhou et al. (2005) (excluding the Country gazeteer and WordNet features), plus several additional carefully-chosen features that have been highly tuned for ACE-style relation extraction over years of research (Sun et al., 2011). The embedding-based features are from Yu et al. (2014), which represent each word as the outer product between its word embedding and a list of its associated non-lexical features. The non-lexical features indicate the word's relative positions comparing to the target entities (whether the word is the head of any target entity, in-between the two entities, or on the dependency path between entities), which improve the expressive strength of word embeddings. We store the extracted relations in CONCRETE `SituationMentions`. See Figure 2 for an example visualization.

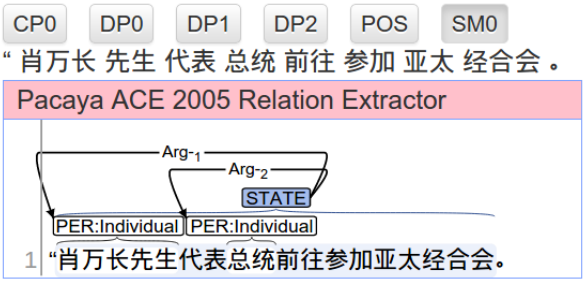


Figure 2: ACE entity relations viewed through Quicklime (Section 3.7).

3.6 Cross Document Coreference Resolution

Cross document coreference resolution is performed via the phylogenetic entity clustering model of Andrews et al. (2014).[5] Since the method is fully unsupervised we did not require a Chinese specific model. We use this system to cluster `EntityMentions` and store the clustering in top level CONCRETE `Clustering` objects.

3.7 Creating Manual Annotations

Quicklime[6] is a browser-based tool for viewing and editing NLP annotations stored in a CONCRETE document. Quicklime supports a wide array of analytics, including parse trees, token taggings, entities, mentions, and “situations” (e.g. relations.) Quicklime uses the visualization layer of BRAT (Stenetorp et al., 2012) to display some annotations but does not use the BRAT annotation editing layer. BRAT annotations are stored in a standoff file format, whereas Quicklime reads and writes CONCRETE objects using the Thrift JavaScript APIs. Figure 1 shows Quicklime displaying annotations on Chinese data. In particular, Quicklime can combine and overlay multiple annotations, such as entity extraction and dependency parses, as in Figure 1b. Figure 2 shows entity relation annotations.

Acknowledgments

We would like to thank the reviewers for their helpful comments and perspectives. We would also like to thank the Johns Hopkins HLTCOE for providing support. A National Science Foundation Grad-

[5] https://bitbucket.org/noandrews/phyloinf

[6] https://github.com/hltcoe/quicklime

uate Research Fellowship, under Grant No. DGE-1232825, supported the second author. Any opinions expressed in this work are those of the authors.

References

Jacqueline Aguilar, Charley Beller, Paul McNamee, Benjamin Van Durme, Stephanie Strassel, Zhiyi Song, and Joe Ellis. 2014. A comparison of the events and relations across ace, ere, tac-kbp, and framenet annotation standards. In *ACL Workshop: EVENTS*.

Nicholas Andrews, Jason Eisner, and Mark Dredze. 2014. Robust entity clustering via phylogenetic inference. In *Association for Computational Linguistics*.

Pi-Chuan Chang, Michel Galley, and Christopher D Manning. 2008. Optimizing chinese word segmentation for machine translation performance. In *Third Workshop on Statistical Machine Translation*.

Linguistic Data Consortium. 2013. DEFT ERE annotation guidelines: Events v1.1.

Francis Ferraro, Max Thomas, Matthew R. Gormley, Travis Wolfe, Craig Harman, and Benjamin Van Durme. 2014. Concretely Annotated Corpora. In *AKBC Workshop at NIPS*.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL*.

Dan Klein and Christopher D Manning. 2003. Accurate unlexicalized parsing. In *ACL*.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *ACL: Demos*.

Courtney Napoles, Matthew Gormley, and Benjamin Van Durme. 2012. Annotated gigaword. In *AKBC-WEKEX Workshop at NAACL 2012*.

Mark Slee, Aditya Agarwal, and Marc Kwiatkowski. 2007. Thrift: Scalable cross-language services implementation. Facebook White Paper.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Sophia Ananiadou, and Akiko Aizawa. 2012. Normalisation with the brat rapid annotation tool. In *International Symposium on Semantic Mining in Biomedicine*.

Ang Sun, Ralph Grishman, and Satoshi Sekine. 2011. Semi-supervised relation extraction with large-scale word clustering. In *ACL*.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL*.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus ldc2006t06. *Linguistic Data Consortium*.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer. 2005. The penn chinese treebank: Phrase structure annotation of a large corpus. *Natural language engineering*, 11(02):207–238.

Mo Yu, Matthew Gormley, and Mark Dredze. 2014. Factor-based compositional embedding models. In *NIPS Workshop on Learning Semantics*.

GuoDong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring various knowledge in relation extraction. In *ACL*, pages 427–434.