Constructing a crowdsourced linked open knowledge base of Chinese history

Donald Sturgeon
Durham University
United Kingdom
donald.j.sturgeon@durham.ac.uk

***Abstract*—This paper introduces a crowdsourced approach to knowledge base construction for historical data based upon annotation of historical source materials. Building on an existing digital library of premodern Chinese texts and adapting techniques from other annotation and knowledge base projects, this lays the groundwork for a scalable, sustainable, linked open repository of data covering around 3000 years of recorded Chinese history.**

Keywords—Linked Open Data, knowledge graphs, semantic web, Chinese history

I. Introduction

Historiographic works, such as the 25 standard histories (正史), form an important part of the historical record of premodern China. These works are a key source of knowledge about many aspects of historical Chinese civilization, including details of people, events, bureaucratic structures, literature, geography, and astronomy. Given the historical importance of these works, many such sources have been digitized; however, in most cases these digitized texts contain only the literal textual content of the work, and therefore remain challenging to process computationally. Similarly, while some portion of the knowledge contained in each source is available in scholarly databases, a much greater amount of the information they contain does not yet exist in any machine-readable form.

This paper introduces a crowdsourced system to efficiently extract historical data from these texts and record it in a reusable, machine-readable form. Based upon an existing widely used full-text digital library of classical Chinese works – the Chinese Text Project (https://ctext.org) [1] – which already contains large numbers of historiographic texts including all of the standard histories, this system enables the creation and management of two additional types of data to facilitate this process: semantic annotations of entity references within texts, and a knowledge graph recording data about these entities and their relationships to one another. A public user interface is provided to facilitate the creation and maintenance of both annotations and knowledge claims at scale through crowdsourcing, which makes use of the current state of the knowledge graph to suggest probable annotations, and uses the current state of the annotations to propose additions to the knowledge graph based on the semantic content of a text.

The system provides a scalable and consistent way for extracting and recording historical data of a variety of types from primary source texts. Initial data consists of over 250,000 claims about more than 50,000 entities. For entities representing people, the vast majority of entities referenced have at least one external authority source (primarily Wikidata, with over 95% coverage), facilitating straightforward integration with other datasets. Lastly, each entity, property, and qualifier is represented by a unique URI, and the knowledge graph itself is automatically serialized into an RDF representation that can be downloaded in bulk under an open license for analysis and reuse.

II. Annotation of text

A. Entity annotation

In order to connect textual content to a knowledge base, some form of annotation is desirable in order to allow precise distinguishing between identically written tokens that in fact refer to entirely different entities due to the contexts in which they appear. A natural way to add annotations to existing textual content is by means of a markup language such as XML; annotations then consist of a pair of opening and closing tags in the markup language which identify a precise span of text and record additional information about the meaning of this span of text within its given context. In the implementation described in this paper, semantic annotations are added to texts which already contain various non-semantic annotations recording structural information about the text, as well as the relationship between parts of the transcribed text in the XML document and page images of the historical work that the digital edition is based upon.

Semantic annotations in general could contain a variety of information about the annotated textual content. For text referencing a person, for example, an annotation would likely at a minimum record the fact that the span of text refers to a person (as opposed to some other type of entity, such as a place). Beyond this, further information to identify *which* person is referenced is also desirable; this might take the form of one or more authority identifiers, or potentially other known information about the individual where no such identifiers are available (e.g. the person was born in a particular place, lived during a particular time period, etc.). Given the very large potential range of information that might usefully be encoded in these semantic annotations even for a single class of entities such as historical individuals – together with obvious issues of unnecessary data duplication if this information were recorded in multiple annotations all referencing the same person – a more practical approach is to record in each annotation only the minimum information required to identify a particular entity using a simple schema that is unlikely to require modification due to future changes in the scope of semantic information to be recorded. To achieve this, in this implementation every semantic annotation supplements the text with two pieces of information: an entity *type* (chosen from a controlled vocabulary, corresponding to a range of core types of entity such as "person", "place", "era", "work", etc.), and a globally unique entity identifier automatically assigned by the system. The purpose of the

entity identifier is to connect each individual reference to the same entity to all other references to that entity, while also linking it to whatever further information is known about the entity (including its names, connections to external authority sources, as well as historical information about the entity, as appropriate to the entity type). All other information about the entity is stored externally to the annotated text as a knowledge graph recording relevant data about entities and their connections to one another. Only one exception is made to this approach: annotations representing historical dates are recorded using additional information stored directly in the annotation in order to simplify the recording of date information and facilitate conversion between calendar systems.

While the annotation data is stored using an XML representation which can be directly edited, most annotations are created by using a point-and-click interface (Fig. 1), which builds upon interface design approaches used in existing annotation systems such as Recogito [2] and MARKUS [3]. Sections of text are loaded from the digital library using its Application Programming Interface (API) and displayed in the main part of the screen. Any existing annotations are loaded as annotations in the "confirmed" state, indicated using solid blocks of color, with the color distinguishing between annotations of different entity types. At this point, the user can create new annotations manually by selecting part of the text using the mouse, or alternatively request that the interface proposes possible new annotations by comparing the contents of the text with the names of known entities. In the latter case, the interface queries the entity database as to what names are defined, using metadata about the date of composition of the text being annotated (where available) to exclude entities and names that are known to post-date the composition of the text being annotated. Using this data, the interface then adds its own annotations to the text, and sets all of these new annotations to the "unconfirmed" state, indicated visually by a solid grey background with a colored underline indicating the suggested entity type.

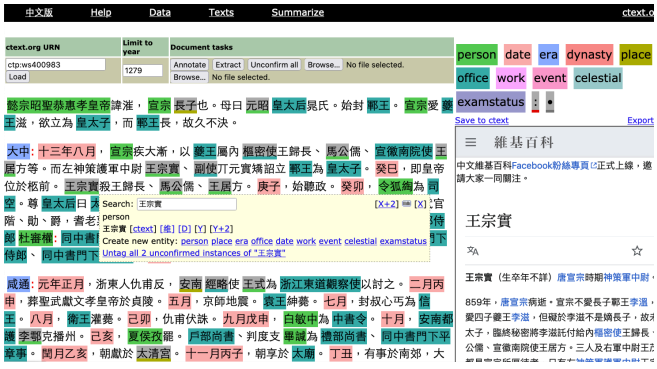

Fig. 1. Annotation interface with source text (left) partially annotated. The unconfirmed annotation suggested by the system (yellow box) is linked to an entity in the system's knowledge graph, Wikidata, and the Chinese Wikipedia article shown on the right in this screenshot; the user can use any of these to determine which entities should be linked.

At this point in the annotation process, the core task of the user is to correctly determine two things: 1) which characters of the text represent mentions of an entity; 2) for each such instance, what is the corresponding entity. Given the chosen data model, the latter step ultimately implies specifying a unique entity identifier (having first created a new identifier for the current instance if this entity has not previously been annotated anywhere else in the corpus). While this process is straightforward conceptually, a key practical challenge faced is how to provide appropriate assistance to the user in the second step of the task. For example, a user might create a new entity e_1 to represent person A_1 in a text, and then a second entity e_2 to represent person A_2, who coincidentally shares the same name as person A_1. When subsequently presented with a third occurrence of this same name, another user would have to examine the previously annotated instances of the two entities to determine whether or not either of them was the same person as the new occurrence. Partly for this reason, external sources of data about relevant entities are invaluable in practical execution of the annotation task. In the case of historical individuals, external sources such as Wikipedia, Wikidata [4], and the China Biographical Database (CBDB) [5] all provide structured and/or unstructured data about relevant entities, together with identifiers that allow disambiguation between distinct same-named entities. While these sources generally structure their data in very different ways, and often use different identifier systems, aligning entities in the system described here with these external sources by recording their identifiers makes it possible for users of the annotation interface to base annotation disambiguation decisions on relevant information included in any of these external sources.

As a result, it is desirable to inform users of the existence of possible matches in external systems during annotation, even where no corresponding entities have been created locally. By doing this, users are encouraged to disambiguate new entities as they are created, meaning that external identifiers can be added to newly created entity records automatically, ensuring that any user faced in future with the task of disambiguation of an entity with this name can similarly benefit from contextual information about the entity stored in these external resources without having to examine the context in which annotations of the entity were previously made. Key to the effectiveness of this task is the role of Wikidata, which maintains mappings between many relevant identifiers in addition to its own, including CBDB identifiers, as well as the entity identifiers used by the system described in this paper. This data makes it possible to combine identifiers from different systems that represent the same entity into a single candidate choice when displaying these to the user, avoiding unnecessary duplication.

When the user is satisfied with any changes made, the system saves the updated XML document back to the digital library, ignoring any annotations still in the unconfirmed state. New entities are created according to the data provided, and their newly assigned entity identifiers added in the appropriate locations in the text.

B. *Date annotation*

While many types of entity – including people, places, bureaucratic titles, etc. – can be adequately annotated by using a single identifier to point to some externally defined entity, references to dates form one particularly important exception to this, and are therefore handled somewhat differently from all other types of annotation. Dates in historical Chinese works are themselves moderately complex, due to multiple ways of expression (e.g. day *n* of a calendar month, vs. day *m* in a continuously repeating sixty-day cycle), as well as large numbers of named points of reference being used. These points of reference are generally either the name of a ruler of a state, or the name of an era as proclaimed by a ruler. Given such a point of reference, precisely specified dates then consist

of a year (either as an ordinal starting at 1, or as year *m* in a continuously repeating sixty-day cycle), a month (including leap-months), and a day. Together with precise information about the point of reference (i.e. which ruler or era), this data is generally sufficient conceptually to specify an historical date precisely. By combining this data with prior work on Chinese historical calendars, it is then possible to convert such a date to the Julian and Gregorian calendar systems [6].

Although conceptually this task is straightforward, there are a number of practical difficulties involved in the process. One important caveat is that historical texts frequently "misuse" dates: a text may refer to the first day of the first month of the first year of an era, even though the era was in fact not declared until much later in the year – i.e. calendars may be used proleptically. For some dates – particularly earlier dates, such as those in the pre-BC era – there may be uncertainty or scholarly disagreement about how to correctly interpret a particular date. In order to avoid these issues from causing problems with the annotation workflow, instead of performing calendar conversion during the annotation process and recording dates in existing formats associated with other calendars (e.g. as a Julian or Gregorian date, or a Julian Day Number), the annotation system instead annotates Chinese dates in a machine-readable form that makes explicit what the literal, contextualized meaning of the date is – i.e. which era, and which year, month, and day within that era. This allows representation of proleptic dates as well as dates that may be incorrect or invalid according to calendar conversion data; all dates that *are* valid can be mechanically converted to the Julian and Gregorian calendars. If errors are subsequently identified in the calendar conversion data, these can be corrected without requiring changes to any annotated texts.

A remaining practical challenge is that in many historical texts, date references are highly contextual. Where many events are recorded in chronological order with dates provided, most commonly those parts of the dates that would be contextually clear to a reader of the text are omitted after their first reference. For example, a text might begin by explicitly mentioning year 13, month 8 of the *Dazhong* (大中) era – a "complete" date (i.e. not missing any data required to interpret the date; in this case the date refers to a period of one month) that can easily be annotated. After mentioning what happened, more specific dates may be mentioned, omitting the era, year, and month – for example, simply reading "*Guisi* 癸巳", i.e. "30th in a cycle of 60". In this case, understanding the meaning of the date requires knowledge of the context, and it is this context which must be determined and explicitly recorded during the annotation process in order to make the date mechanically translatable to other calendar systems. In other contexts, an identical expression could also be used to refer to a specific year (rather than a day), i.e. the 30th *year* in a repeating sixty-year cycle. While the contextual flow of date information is straightforward in this example, in general it is non-trivial due to parenthetical references to other dates which do not imply that subsequent dates in the narrative should be read as being relative to the parenthetically mentioned date.

In order to facilitate systematic and precise recording of historical dates and enable automatic calendar conversion, the annotation interface provides users with assistance in identifying and recording date annotations. These record a date as a combination of an entity representing either an era (such as the *Jianyuan* 建元 era of *Han Wudi* 漢武帝) or a ruler (such as *Qin Shi Huang* 秦始皇) and optionally the year, month and day within the Chinese calendar system (such as year 1, month 4, day 58 of the sixty-day cycle). The user interface implements simple rules for "flowing" date information through a text, so that the contextually implied values for eras, years, and months are preselected by the system, allowing efficient annotation of dates. By leveraging data created by Dharma Drum's Time Authority Database [6], these annotations make possible fully automatic real-time conversion of dates into the Julian and Gregorian calendars, precise to the day.

Lastly, dates can be serialized into a regularized format that represents their semantic content, rather than their interpretation in a different calendar system. This simply records in sequence the key pieces of information that make up the semantics of a complete Chinese date: era/ruler (expressed as an entity identifier), followed by year, month, and day. These machine-readable dates can then be used elsewhere in the system wherever dates need to be recorded, allowing for precise recording of dates as indicated in historical sources, without the recording process itself requiring conversion of dates to a non-Chinese calendar.

III. KNOWLEDGE GRAPH

A. Data model

As texts are annotated with references to entities, it becomes useful to record data about these entities. This is particularly important because a substantial proportion of entities referenced in these historical texts are entities that are relatively obscure in modern terms – such as those which, though once sufficiently notable to be mentioned (perhaps only in passing) somewhere in a standard history, have left little else in the way of concrete information in the modern historical record. This is the case, for instance, with many individuals who are mentioned as having a kinship relationship with an individual to whom much more space in the historical record is devoted. At the same time, a small but important subset of the entities represent the "famous actors" of historical China: the rulers, ministers, rebels, generals, and scholars who historiographers considered to play important roles in history, and who therefore are the subject of more substantial records in the histories themselves – very often in addition to substantial historical information recorded elsewhere. Naturally, many of these people remain important and notable subjects of study today, and as such frequently have dedicated pages in encyclopedias such as Wikipedia, which can be leveraged productively in the annotation process. However, for the many entities that are have neither encyclopedia pages nor entries in scholarly databases like CBDB, references to external resources cannot be used to assist in disambiguation during the annotation task. Instead, data about entities identified in texts is recorded locally in a structured, versioned knowledge base, and this knowledge base used to assist in subsequent annotation.

Given the substantial complexity involved in recording historical data, a key design requirement for recording this information is that modifications can be made in future to the type of data recorded without requiring any changes to the program code or method of data representation. In order to achieve this goal with minimum code complexity, a graph-based approach is used, in which data about an entity is stored exclusively as a set of verb-object claims, all of which have that entity as their subject. The conceptual model used closely follows the structure of Wikidata, and from a user perspective

consists primarily of "entity records", each of which collects data about a particular entity such as a person, a literary work, or a place. Apart from an automatically assigned identifier used to refer to the entity and connect it to mentions of the entity in annotated texts, entity records consist entirely of a set of claims about the entity. Each claim consists of a subject (the entity to which the claim applies), a property (i.e. verb) chosen from a controlled vocabulary defining the types of property applicable to entities of a particular type, and an object, which may be either another entity, a machine-readable date, or a literal value such as a string or integer. Claims may also have qualifiers which represent qualification about what is being claimed, again consisting of a verb chosen from a controlled vocabulary and a target object – for example, to indicate that the claim only applies starting from (or up until) some particular date. Each claim and qualifier may also have a textual citation serving as evidence for the claim or qualification; these citations are stored in a machine-readable format, and are added automatically when knowledge claims are input or extracted through the annotation interface. By means of the existing functionality of the Chinese Text Project, these citations additionally provide a method to locate the specific line of text cited on the appropriate page of a scanned copy of the edition of the text upon which the digital transcription is based.

The list of valid properties and qualifiers is dynamically maintained, and itself forms part of the knowledge graph (Fig. 2). It can be edited through the same interface as other entities, and all other components of the annotation and knowledge graph system use this information – rather than hard-coded values – in determining what vocabulary can be used, and in what ways.

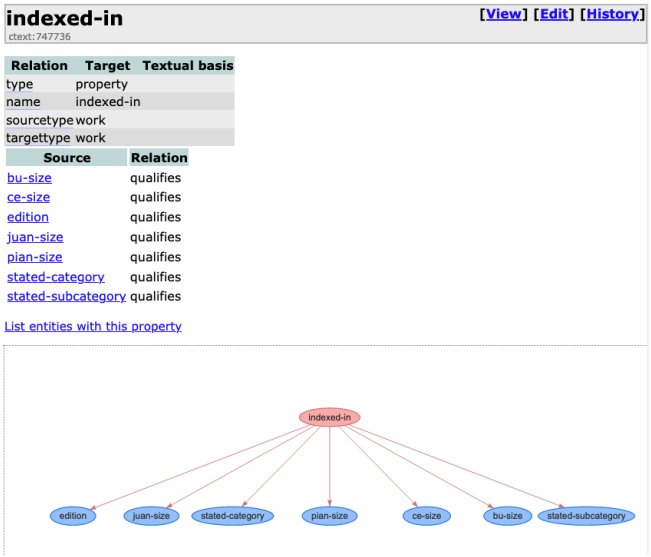

Fig. 2. Entity record for the property "indexed-in". This record indicates that this property can only be used to connect an entity of type "work" to another entity of the same type, and that this property has a number of possible qualifiers (each of which is itself an entity of type "qualifier", and similarly contains data defining how that qualifier may be used).

B. Editing

Like annotations, data in the knowledge graph is versioned, crowdsourced, and can be edited directly by users. As with annotations, while it is possible to edit the knowledge graph directly by modifying the contents of its underlying serialization, it is far more intuitive and practical to edit it through a purpose-designed interface providing assistance with the task. Whereas annotation in general begins in the first instance with a sequence of unannotated characters, knowledge graph construction can leverage semantic annotations that have been added to a text. In many historical texts, certain expressions are repeatedly used in substantially similar or formulaic ways to record similar types of information about different entities. As a simple example, many historical texts include biographies of large numbers of people, and many of these begin with a statement of the person's style name and place of origin – e.g. "蘇軾，字子瞻，眉州眉山人。" (Su Shi, styled 'Zizhan', a person from Meishan, Meizhou). In some cases, patterns like these can be extracted directly from unannotated text with a high degree of precision using regular expressions. However, this task becomes significantly easier once texts have been annotated, because annotation types can be taken into account when looking for patterns. For example, bureaucratic offices and titles are extensively recorded in historical texts, with one commonly used way of recording that a person took up an office or was awarded a particular title using the verb "為" (*wei*, to make), such as "己未，王繼英為樞密使。" (On day 56, Wang Jiying was made Commissioner). The verb "*wei*" is extremely common in literary Chinese, and it would be a challenging task in general to accurately distinguish cases where it indicates that a person took up an office or title from cases where it means something entirely different. However, given a correctly *annotated* text with the same content, this task becomes far simpler: if instead of looking for strings that happen to contain this common character we look only for strings with the form "<date>，<person>為<office>。" it becomes very easy to identify precisely those cases where a text claims that a person took up an office or title.

Based on this approach, the annotation client provides three methods of adding data to the knowledge graph. The first of these is automated extraction from an annotated text. By applying a sequence of regular expressions, candidate sources of historical data are identified in the text. Depending on the particular expression, these may match the literal content and/or general entity type in the annotated text. Each regular expression is paired with a machine-readable description of the claim or claims that generally follow from a statement of the given form. For instance, in the above example, the claim that follows would be that "<person> held-office <office>", and this claim would be qualified with the qualifier "from-date <date>". Having conceptualized these suggested claims, the annotation client then queries the knowledge graph to check whether these claims about the subject entity (here, the person in question) are already included in the knowledge graph, and indicates this visually to the user (Fig. 3). If the user accepts the claim, it is immediately added to the knowledge graph, and the highlighted text matched by the regular expression is stored as a machine-readable citation justifying the addition of this claim.

Fig. 3. Automatic knowledge claim extraction. The first set of claims extracted (blue) are already recorded in the knowledge graph; the second set

(red, line 2, expanded in the box below), corresponding to the text "Day 24, Wang Rong killed Li Kuangwei", are not. Accepting the two suggestions proposed will add a claim to the entity representing Wang Rong, indicating that he killed Li Kuangwei on this date (May 8, 893), and a claim to Li Kuangwei's entity record indicating that he died on this date.

Knowledge claims can also be added manually by the user through the annotation interface. Since most knowledge claims require textual evidence (exceptions include names of entities, authority identifiers, etc.), this is done by first selecting the region of text that justifies the claim to be added. The system then uses a set of heuristics to propose which entities are likely to be subjects of a claim justified by this piece of text – likely candidates include annotated entities occurring within or prior to the selected region of text, as well as those frequently mentioned in the text as a whole. Having selected a subject, the interface then offers a list of verbs that can be validly applied to a subject of that type according to data about properties and qualifiers stored in the knowledge graph. When one of these is selected, the interface checks whether there are any entities or dates referenced within or slightly before the selected region of text which are of a type allowed as the object of the selected verb, and if so, these are displayed so the user can select one of these quickly; alternatively, the user must type in the appropriate object, and save the claim.

Lastly, particularly for large texts with fixed or formulaic structures, fully automatic annotation can also be integrated into the process. In this case, annotations are created according to task specific rules and/or user supplied data; the annotation client applies these rules to the text in bulk to produce a preview of changes to be made, which can then be accepted and applied through a single operation.

C. *Querying and data export*

The knowledge graph is most straightforwardly navigated by hyperlinks which connect textual content to entities, and entities to other entities with which they share an edge. Minimal task-specific processing is performed in displaying entity records, mainly consisting of sorting edges according to type and suitable qualifier values (e.g. sorting appointments by their "from-date" qualifier, so that offices held are ordered chronologically when displayed). Relationships that are hierarchical (such as the "father" edge type) are visualized as interactive trees in order to provide better contextualization; similarly, edges connecting an entity to geographical data for which coordinates are known are visualized on a map.

Basic querying can be accomplished within the web interface by specifying edge types and value ranges (either as strings or entity identifiers). Annotated texts can also be used as query terms via their Uniform Resource Names [1], allowing queries requesting lists of entities matching particular criteria in the knowledge graph that occur (or alternatively, do not occur) in one or more texts.

The knowledge graph can be exported as a Resource Description Framework (RDF) serialization, either at an individual entity instance level, or as a bulk download produced through regular automated snapshots [7]. All entities, dates, properties, and qualifiers have persistent Uniform Resource Identifiers (URIs), enabling their use externally in semantic web applications. RDF dumps can be imported directly into triple stores for querying using SPARQL and similar query languages. The existing API providing access to textual data from the Chinese Text Project has been extended to further enable access to both annotated texts and the knowledge graph, facilitating close and real-time integration with other projects. Notably, the annotation interface itself is implemented as a client-side application which communicates with the digital library as an API client, rather than being a core component of the digital library infrastructure. As a result, it is also possible to use the client to annotate external, user-provided texts, without requiring that these first be stored within the digital library itself.

IV. Integrating Data into the Library

Semantic annotations have many obvious applications in modern digital libraries and full-text search systems. Particularly with historical materials, access to additional contextual information can greatly improve the accessibility of a text to readers less familiar with its subject matter. The most straightforward implementation of this is achieved by linking entity mentions in texts to data about the entities referenced. This approach is used for all texts with semantic annotation on ctext.org, with some additional enhancements for convenience of reading. Annotation types are indicated visually through colored underlines; references to entities display a preview window when hovered over in a text (Fig. 4); date annotations display the corresponding date or date range in the Julian or Gregorian calendar and the explicit (contextualized) date information in Chinese, and link to complete calendars for the corresponding era or ruler.

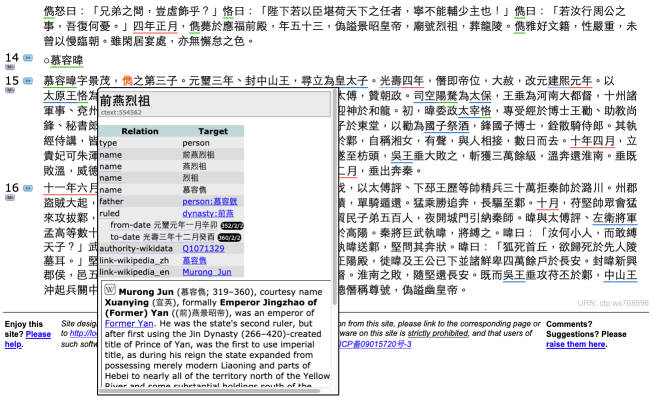

Fig. 4. An annotated text as displayed in the Chinese Text Project digital library.

Many other types of reading assistance are also made possible by the combination of semantic annotations and the knowledge graph. Given the close alignment to entries in Wikipedia, one such enhancement consists of leveraging Wikipedia's encyclopedic content to further contextualize those entities for which encyclopedia entries exist. While the Wikipedia articles themselves can be easily accessed by following the hyperlink from the corresponding entity record, a closer form of integration has also been implemented. By reparsing the "wikicode" serialization of text in which articles are stored in Wikipedia, and connecting all appropriate linked words and phrases in the text to the knowledge graph via their Wikipedia URL, it becomes possible to automatically create human-readable summaries of important entities which also enable intuitive navigation of the knowledge graph. Whereas following links in the Wikipedia article would take the user to another page of Wikipedia, after integration with the knowledge graph these links take the user to a combination or "mashup" of entity data and links to mentions of the entity in historical texts, together with Wikipedia's narrative encyclopedia entry for the entity (where one exists in an

appropriate language). Though the knowledge graph itself contains no English data other than labels for properties and qualifiers, by combining the linked data from Wikidata and Wikipedia, a complete, readable English-language summary can be produced for many thousands of entities. Thanks to previous work done by the Wikidata and Wikipedia communities to connect pages across different language versions of the encyclopedia, equivalent summaries can easily be created for any language for which sufficient encyclopedia coverage is available. Currently over 10,000 entities (mostly historical people) are aligned to Chinese Wikipedia through this process.

Alongside reading, full-text search is a fundamental application of digital libraries and full-text database systems, and again is a task for which straightforward but powerful enhancements are made possible through semantic annotation. Most obviously, it becomes possible to search for mentions of an entity in a text regardless of what name it is referred to by – something particularly useful in the case of Chinese historical texts, where the surname of a person may frequently be omitted in many contexts where it would be obvious to the reader, and the personal name alone may consist of a single common character that is often used in other contexts where it does not refer to any person at all. Similarly, the ability to search for references to particular dates, or dates within a particular range, irrespective of the form in which the date happened to be recorded, provides a more intuitive way of searching than is achievable without semantic annotation.

Many further incremental enhancements of this approach are made possible by combining full-text search with knowledge graph search. For example, one could query for all textual references to geographic places *within* a particular region, or all mentions in a given text of anyone having some familial relationship to a specific person.

One aspect of the data encoded in the knowledge graph particularly relevant to the digital library itself is data about historical works and their authors. By modelling both historical works and authors as entities, and including relevant identifiers such as VIAF (Virtual International Authority File) for people for whom such identifiers exist, a flexible means of recording authorial data for historical works is made possible within the more generally applicable data model used by the knowledge graph. Connections between entities representing works and sets of particular editions of texts are easily made as edges in the knowledge graph. This approach has the considerable advantage in metadata processing of complete consistency in handling works that exist in the digital library, works that only exist elsewhere, and works that are no longer extant: having one or more editions of a text in the digital library is simply a property of the entity record for that work, which links it to the specific digital copies.

V. Conclusions

This paper has introduced a scalable approach to crowdsourced knowledge base construction for historical Chinese sources. By reusing and building upon existing infrastructure, the system has been made widely available to an engaged community of users and editors who have already demonstrated a willingness to contribute to improving digital editions of texts through crowdsourcing. While substantial numbers of annotations and knowledge claims have been created, far more remains to be done. As the scope and completeness of both annotations and entity data grow over time, it is hoped that this dataset will become an increasingly useful tool for the quantitative study of aspects of Chinese history and historiography.

References

[1] D. Sturgeon, “Chinese Text Project: A Dynamic Digital Library of Pre-modern Chinese”, Digital Scholarship in the Humanities 2021, 36(s1).

[2] R. Simon, E. Barker, L. Isaksen, and P. Cañamares (2015). “Linking early geospatial documents, one place at a time: annotation of geographic documents with Recogito” e-Perimetron, 10(2), pp. 49–59.

[3] H. De Weerdt, “Creating, Linking, and Analyzing Chinese and Korean Datasets: Digital Text Annotation in MARKUS and COMPARATIVUS”, Journal of Chinese History, 4(2), pp. 519–527.

[4] D. Vrandečić and M. Krötzsch. “Wikidata: A Free Collaborative Knowledgebase”, Communications of the ACM, 57.10 (2014), pp. 78–85.

[5] L. Tsui, and H. Wang, “Harvesting Big Biographical Data for Chinese History: The China Biographical Database (CBDB)”, Journal of Chinese History, 4(2), pp. 505-511.

[6] M. Bingenheimer, J. Hung, S. Wiles, and B. Zhang. “Modelling East Asian Calendars in an Open Source Authority Database”, International Journal of Humanities and Arts Computing 10.2 (2016), pp. 127–144.

[7] https://ctext.org/tools/linked-open-data