Fred's Reusable Evaluation Device: Providing Support for Quick and Reliable Linguistic Annotation

Hannah Copperman, Christopher R. Walker
Microsoft: Bing Search (Powerset)
475 Brannan St, San Francisco, CA 94107
hannahc@microsoft.com, chriwalk@microsoft.com

Abstract

This paper describes an interface that was developed for processing large amounts of human judgments of linguistically annotated data. Fred's Reusable Evaluation Device ("Fred") provides administrators with a tool to submit linguistic evaluation tasks to judges. Each evaluation task is then presented to exactly two judges, who can submit their judgments at their own leisure. Fred then provides several metrics to administrators. The most important metric is precision, which is provided for each evaluation task and each annotator. Administrators can look at precision for a given data set over time, as well as by evaluation type, data set, or annotator. Inter-annotator agreement is also reported, and that can be tracked over time as well. The interface was developed to provide a tool for evaluating semantically marked up text. The types of evaluations Fred has been used for so far include things like correctness of subject-relation identification, and correctness of temporal relations. However, Fred's full versatility has not yet been fully exploited.

1. Introduction

An interface must fulfill several requirements in order to be useful for evaluating large quantities of linguistic data using human judges. It is necessary to have a streamlined interface for the judges to use. The interface must also accurately represent the data being evaluated. And last, the curators or consumers of the data must be able to interact with the data and the judgments clearly and easily. In an effort to meet these three requirements, we have created an evaluation interface for what we call SemXML (semantically marked up sentences).

SemXML is the final product of a natural language content-processing pipeline (Crouch and King, 2006). First, the data (which in the past has usually been sentences extracted from Wikipedia articles, although we have processed other text, such as articles from the New York Times and the Wall Street Journal) is processed by the automated name tagger. Then it is fed into the parser, which returns parse trees. These trees, along with the data from the name tagger, are sent to the semantics component of the system. The resulting semantics markup includes things like thematic roles (such as agent, patient, actor, et cetera), temporal and locative relationships (i.e., "Lincoln::WHR::Ford's Theater), frame alternations ("Pat married Tony", "Pat and Tony married"), and various other types of semantic information.

To evaluate the SemXML (Walker et al., 2010) we extract sentences containing some amount of semantic markup. For instance, the WHN evaluation task contains sentences containing a WHN (temporal) relation. These sentences are presented to a judge, who is asked to decide whether the relation in question is correctly assigned. For example, the following is an instance of a correct WHN relation:

They left on Tuesday.
leave::WHN::*Tuesday*

Each evaluation task consists of at least 1000 sentences, and each task is performed by at least two judges. In addition, we evaluate two data sets: a stable set (this set always consists of the same articles, processed by each new version of the NL pipeline) and a random set. Since we currently do 12 different evaluation tasks for each new version of the semantics system, every evaluation requires (at least) 48,000 individual judgments. The interface must therefore support human judges doing such a large volume of judgments. It must also let those who will use the data (for development, evaluation, or testing) compare different versions of any particular evaluation, as well as do other, perhaps larger, comparisons, such as comparing the precision of a particular evaluation over time, comparing evaluation type against evaluation type, or comparing precision numbers for different annotators, and so on.

The evaluation interface we designed to meet these needs is called Fred (which stands for "Fred's Reusable Evaluation Device"). Fred is currently used to support all of our SemXML evaluations, as well as some other similar evaluations. It is also extensible and fairly versatile. What follows is a description of Fred, and a discussion of potential applications and future work for Fred.

2. Technical details

Fred is built in the Ruby on Rails framework (rubyonrails.org), hosted on a machine on our internal

network. Our judges, who are all offsite contractors, can access the machine through a remote network connection.

2.1 Administrative Side

In order to create an evaluation task on Fred, an administrator must upload a file that contains the data to be evaluated. Once the file has been uploaded, the administrator can choose the type of evaluation appropriate for the data; that is, the administrator decides what judgment choices should be presented to annotators. Currently, the evaluation type we use most often is "Correctness". This evaluation type provides the annotators with three choices: "Correct", "Incorrect", and "Unjudgable". After selecting the judgment type, the administrator submits the task. Submitting a task makes it available for annotators to judge.

2.2 Annotation Side

When the administrator submits a task, Fred makes that task available on the annotator task-claim page. Annotators can claim tasks by clicking the "Claim" button associated with a particular task. Each task can be claimed by no more than two annotators. Once an annotator claims a task, they can come back to it at any time. The sentences to be evaluated are presented one per line, with radio buttons for judgment choice. The "Submit" button will submit only those sentences that have been judged; any left blank will be re-displayed at some later time to the annotator (the sentences are displayed in random order). Annotators can leave the tool at any time; their progress will be saved, and when they return to the tool they can recommence the task that they had been working on. Figure 1 shows an example of an object judgment ("OB"), as it would be presented to an annotator (note attribute tables and judgment options).

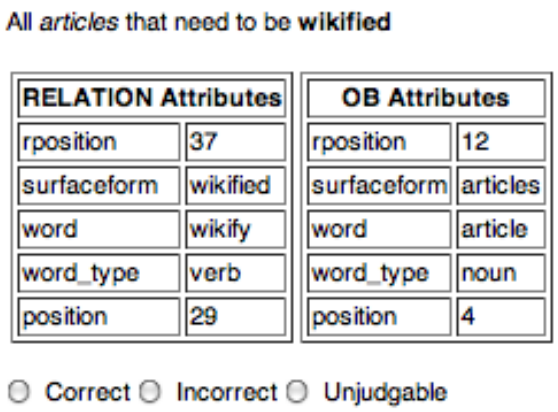

Figure 1: Example OB judgment in Fred

2.3 Data Format

When an administrator uploads a new file to Fred, that file must conform to a specific format. Fred will treat each line as a "sentence" (the terminology comes from our current usage; it doesn't have to be an actual sentence) and will provide judgment options for it. There can be more structure within each line, as well. If fields within a line are separated by tabs, Fred will interpret them as follows:

1st field: judgment value (e.g. *Correct* or *Incorrect*)

2nd field: sentence or document content for display

3rd to penultimate fields: attribute table(s)

Last field: judgment ID

The judgment value is blank when the document is first prepared to be uploaded into Fred. After all of the examples have been evaluated, the data can be exported to the same tab-delimited format, but now with the first field populated by a judgment value. This allows for the administrator to do more complex data analysis and manipulation than Fred currently supports.

The attribute table field can be empty, or it can have as many values as desired. Currently, these attribute tables are used in our SemXML evaluation program to display information such as word type (i.e., noun, verb, et cetera), derivation path (i.e., "garden" (verb) from "gardener" (noun)), or name type (person, location, et cetera). However, the contents of the attribute table or tables are in no way dictated by Fred. There is absolutely no constraint at all upon them. Attributes of any type, linguistic or not, can be placed in these tables. A typical line in a SemXML evaluation source document for the 'subject' (SB) evaluation task would have the following tab-separated fields:

Field1: nil

Field2: His verses tell how he disliked the bustle of the capital

Field3:
role:SB::word:capital::derived_word:nil::word_type:noun::position:70::rposition:77::surfaceform:capital::provenance:nil

Field4:
role:RELATION::word:bustle::derived_word:bustle::word_type:verb::position:56::rposition:62::surfaceform:bustle::provenance:nil

Field5: index_sample/---gNaiKg5YYJ8223lTRgk==.fact::13::109::130573::130718::SB

In this case, there are two attribute tables, one for the SB (field3: *capital*) and one for the RELATION (field4: *bustle*), which is a "derived_word" in this example. The final field contains the source document ID, as well as token offset information, (which may be used by developers for future data analysis or documentation).

3. Results and Other Metrics

Fred makes various metrics available for every evaluation. When the judgment type is "Correctness", one can look at the precision score of the semantic output (as judged by any particular annotator). Always available, no matter the judgment type, is IAA (inter-annotator agreement). The sentences upon which annotators agreed are available to view as a batch, as well as those they did not agree on.

For any set of evaluations there are also certain metrics available. At a glance, one can see how a large evaluation set is faring (for precision as well as IAA), as these metrics are collected in tables that summarize all the numbers for a given data set (Figure 2). Longitudinal tracking of inter-annotator agreement numbers is also available. One table displays an overview of different types of inter-annotator agreement numbers (e.g., chronologically, per evaluation task and per data set). In addition, more in-depth metric analyses are also available, should data users want to look at any individual annotator's inter-annotator agreement numbers over time or per evaluation.

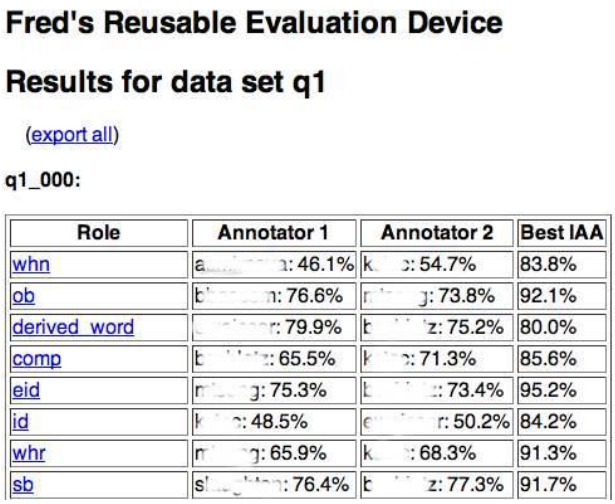

Figure 2: Example results table in Fred, with precision scores for individual annotators, and inter-annotator agreement numbers.[1] [2]

4. Other Applications

As described above, Fred is closely tailored to the particular needs of SemXML evaluation. However, even within the current framework there is room for other projects to use Fred. For instance, sentences to be judged may contain no attribute tables, or they may contain as many as one wants. Judgment types are customizable as well, and there is no limit to the number of values you may add, which means Fred could also perform a more annotation-like role, rather than strictly an evaluation role. Fred has proven to be versatile enough for other evaluation projects besides SemXML. Using Fred we have evaluated system output for automatically generated summaries and paraphrases (Figure 3) and for extracted entity-relation triples (Figure 4).

For paraphrases, annotators were instructed to judge the correctness of the proposed paraphrase, given the source sentence.

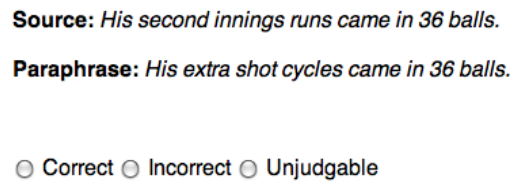

Figure 3: Example of a paraphrase judgment in Fred.

The relation triples task, on the other hand, asks judges about specific semantic relations found in the provided sentences:

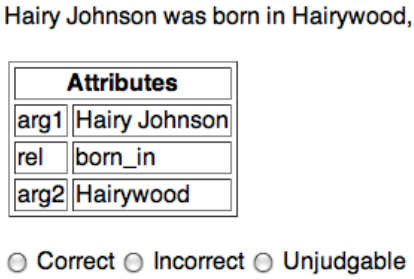

Figure 4: Example of a triples judgment in Fred

Fred's other features are designed specifically to support the creation of high-quality and high-volume linguistic evaluation data. Dual annotation with inter-annotator agreement reporting is absolutely essential if one wants to create reliable data (Uebersax, 2009), and Fred supports dual annotation with no additional requirements for the administrator (cf. Figure 2).

Fred also has built-in functionality for data consumers (rather than data creators) such as inter-annotator agreement tracking that is crucial for annotation or evaluation projects. Administrators being able to indicate evaluation sets to compare synchronically means that Fred will take care of all the tracking details necessary. It also means that evaluation sets are ready for longitudinal comparison without any extra work on the administrator's or data consumer's side.

5. Future Work

While Fred is currently versatile enough to support a variety of different types of evaluation, there are many ways in which it could be improved.

The most crucial improvement is to provide more data analysis tools. Currently Fred data consumers must do all

[1] The "Role" column here lists the individual evaluation tasks, which in this case are all semantic or syntactic roles.

[2] The identities of specific annotators have been blurred for anonymity.

in-depth data analysis offline, using exported data in the tab-delimited format described above. The first improvement to be made should be to enable administrators to slice data and report metrics by different features than the ones mentioned before. For instance, it should be possible for Fred to report precision by sentence length within an evaluation, or report inter-annotator agreement change across evaluation sets for the same evaluation type.

Other features that are more specific to the evaluation task being done would also be desirable, although it might be a bit more difficult to implement. For instance, the ability to specify particular linguistic features, and have Fred display numbers for sentences that contain those features, would significantly reduce the amount of post-hoc analysis data consumers would have to do. In fact, adding that functionality to Fred would practically eliminate the need for any post-hoc data analysis. That functionality would allow a user to query for sentences that contain (for example) prepositional phrases across all evaluation tasks, and draw any conclusions from that data, rather than having to go through each evaluation task individually and process the data at a much lower level.

Other changes are not so crucial, but would definitely improve functionality. An interesting experiment would be to test out different interfaces for presenting judgments to annotators to see how we can increase throughput and whether different presentations of judgments had any effect on inter-annotator agreement. This experiment may become more important as the type of tasks Fred deals with becomes more disparate.

6. References

Crouch, D., and King, T.H. (2006). Semantics via F-Structure Rewriting. *Proceedings of LFG06*, CSLI On-line publications, pp. 145-165. http://cslipublications.stanford.edu/LFG/11/lfg06crouchking.pdf

Ruby on Rails. http://rubyonrails.org/

Uebersax, John. Statistical Methods for Rater and Diagnostic Agreement. (2009). http://www.john-uebersax.com/stat/agree.htm

Walker, Christopher R. and Copperman, Hannah. (2010). Evaluating Complex Semantic Artifacts. In *LREC 2010: Seventh International Conference on Language Resources and Evaluation.*