Introducing Orbis: An Extendable Evaluation Pipeline for Named Entity Linking Performance Drill-Down Analyses

Fabian Odoni
University of Applied Sciences Chur, Switzerland
fabian.odoni@htwchur.ch

Adrian M. P. Braşoveanu
Modul Technology GmbH Vienna, Austria
adrian.brasoveanu@modul.ac.at

Philipp Kuntschik
University of Applied Sciences Chur, Switzerland
philipp.kuntschik@htwchur.ch

Albert Weichselbraun
University of Applied Sciences Chur, Switzerland
albert.weichselbraun@htwchur.ch

ABSTRACT

Most current evaluation tools are focused solely on benchmarking and comparative evaluations thus only provide aggregated statistics such as precision, recall and F1-measure to assess overall system performance. They do not offer comprehensive analyses up to the level of individual annotations. This paper introduces Orbis, an extendable evaluation pipeline framework developed to allow visual drill-down analyses of individual entities, computed by annotation services, in the context of the text they appear in, in reference to the entities specified in the gold standard.

KEYWORDS

Named Entity Linking; Evaluation; Drill-down Analysis

ASIS&T THESAURUS

Evaluation (Term ID: 225)
Natural Language Processing (Term ID: 1246)
Entity Extraction (Term ID: 1152)
Data Visualization (Term ID: 3010)

INTRODUCTION

In order to measure and improve Named Entity Linking (NEL) performance, gold standards are used to evaluate the annotator predictions by using binary classification measures such as precision, recall and F1-measure. Projects and challenges like GERBIL and TAC-KBP provide researchers with powerful tools to determine aggregated performance metrics but do not supply any sophisticated means for visualizing individual results and performing drill-down analyses. They might provide options to inspect single annotations but do not offer intuitive, output formats that can be processed efficiently by humans. Instead, their output formats such as CSV files or spreadsheets contain not much more than the classification status of each annotation (recognized and linked correctly or not). Additional data to investigate the cause of a prediction, for example the context of the annotation in form of the full text or sentence, is not provided. Without this additional data, potential external error causes such as gold standard flaws that were introduced building the data set, Knowledge Base (KB) errors, or the use of different annotation guidelines are hard to detect (Brasoveanu, Rizzo, Kuntschik, Weichselbraun, & Nixon, 2018). Confronted with these problems we started to develop Orbis.

Orbis is a versatile framework for performing NEL evaluation analyses. It supports standard metrics such as precision, recall and F1-score and visualizes gold standard and annotator results in the context of the annotated document. Color coding the entities allows the experts to quickly identify correct and incorrect annotations and the corresponding links to the KB that are also provided by Orbis. Due to the modular pipeline architecture used by Orbis different stages in the evaluation process can be easily modified, replaced or added.

Results of our first Orbis based drill-down analyses efforts were presented at the SEMANTiCS 2018 Conference in Vienna (Odoni, Kuntschik, Braşoveanu, & Weichselbraun, 2018). Motivated by the positive feedback we received for our novel way for answering the why-question, we continued development, to be able to release Orbis as open-source software to a broader user base.

RELATED WORK

Most current evaluation systems focus on measuring performance and thus emphasize aggregated metrics such as precision, recall and F1-score. They offer little to no output to allow drill-down analyses or error analyses.

Tools like GERBIL can be used to evaluate different NEL systems on a large scale (Röder, Usbeck, & Ngonga Ngomo, 2017). GERBIL has been used in many NEL challenges (e.g., ESWC's Open Knowledge Extraction Challenges, ISWC's Semantic Web Challenges) providing more than 10 datasets in its online version, supporting NIF (Natural Language Processing Interchange Format) for gold standards, and multiple experiment types. GERBIL focuses on comparing results between different NEL systems and provides a comprehensive benchmarking system to do so.

82nd Annual Meeting of the Association for Information Science & Technology | Melbourne, Australia | 19–23 October, 2019

DOI: 10.1002/pra2.00049

The scorer (Hachey, Nothman, & Radford, 2014) used in the TAC-KBP challenges (Ji, Nothman, Dang, & Hub, 2016) provides the researcher with an option to inspect individual results and evaluation runs. It yields an output with each predicted entity and the corresponding score (correct, incorrect, missing or false positive) but not the context of the predicted annotation.

APPROACH

Orbis provides an extensible framework to create evaluation pipelines using YAML configuration files for individual evaluation runs. It offers multiple evaluation modes, parallel evaluation runs, resource versioning, and dataset normalization. A basic graphical drill-down analysis is provided by creating html files that can be viewed in the browser.

System Architecture

Orbis consists of two core components (see Figure 1). The first component is the "Orbis Control" component which handles the evaluation management. It loads the evaluation configuration files and executes the evaluations within the pipeline. Depending on the settings it will load all configuration files from a defined folder or just one specific file. Additional data used for multiple evaluation runs will be loaded and cached in order to avoid multiple and thus time consuming reloads.

The second component is the Orbis pipeline. The pipeline consists of three stages and multiple pipelines can be run simultaneously. The first stage is for the acquisition and normalization of the NEL annotations, the second stage handles the assessment of the annotations and the third stage concerns storage and display of the results.

In order to traverse the pipeline an object is created to store all the data and results needed for that evaluation run and provides methods for the separate stages to read and load data. This object is then passed from one stage to the next providing and collecting everything needed for the evaluation.

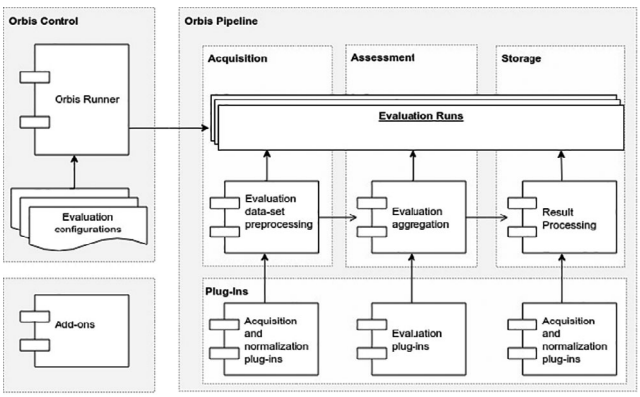

Figure 1. The Orbis system diagram.

The acquisition and normalization stage loads the evaluation corpus as well as its gold standard. Additionally it will search the specified annotation module defined in the YAML configuration file of that evaluation run, load it into Orbis and query the annotation service. A built-in caching framework can be used to speed up evaluations and reduce the load imposed on third party web services. At the time of writing, the following annotation tools are supported by Orbis: Spotlight,[1] AIDA[2] and Babelfly[3] as well as the Recognyze[4] annotation tools.

The assessment stage evaluates the predicted results against the gold standard. The predicted annotations are scored according to scoring rules and the resulting confusion matrix is used to calculate micro and macro precision, recall, and F1-measures.

In the storage and display stage the data is converted to storage and display formats. At this point the evaluation run object contains all the relevant data and the storage stage transforms this data into appropriate formats. Results can be saved as CSV, JSON or even sent to a Graphite server for time series visualization. For a detailed drill-down analysis, and one of the main drivers in developing Orbis, the storage stage can also convert the results into a detailed HTML based, per item view of the predicted entities, embedded in the corpus alongside the gold annotations (see Figure 2).

Plug-in System

Initially Orbis was only designed for internal use and thus was more of a compilation of scripts performing specific tasks. With time, more and more use cases arose and Orbis continually grew to a point where structuring Orbis into a pipeline made sense in order to keep Orbis maintainable. For this reason a modular system was developed to allow Orbis users to define the acquisition, assessment and storage stage components directly in the evaluation run configuration. By copy-pasting the configuration run files and merely changing the type of annotation service the same evaluation of two different annotator services can be run and compared with ease since Orbis would automatically search and load the necessary modules.

To fully utilize this pipeline extensibility, a modularization architecture, based on plug-ins, was implemented providing standardized data loading and storing methods, and devising rules on how a module should be built in order for Orbis to find and load a module. With this plug-in system third parties can develop and implement their own modules to provide Orbis access to additional web services, implement new scoring rules and metrics as well as new output formats.

[1] https://www.dbpedia-spotlight.org/
[2] https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/ambiverse-nlu/aida/
[3] http://babelfy.org/guide#Disambiguateatext
[4] https://www.weblyzard.com/recognyze/

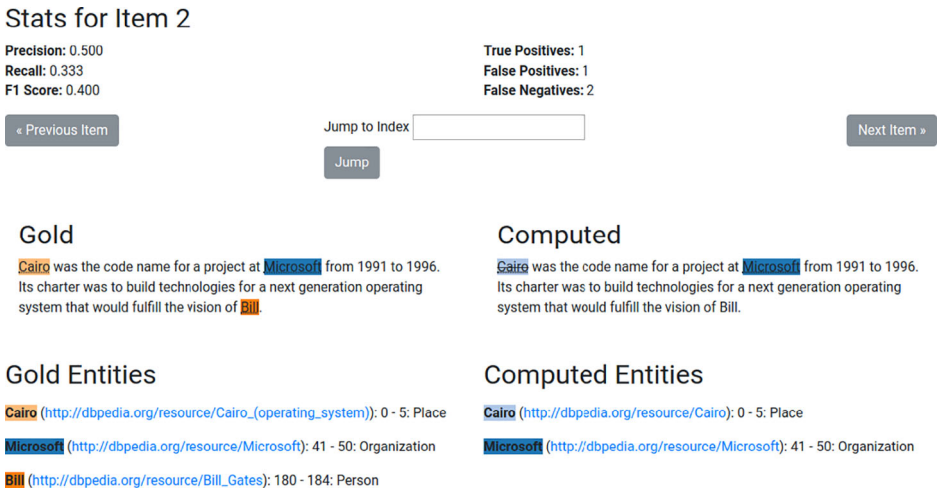

Figure 2. HTML view for drill-down analysis of predicted annotations compared to gold standard annotations.

Monocle

Monocle for Orbis provides a system to manage knowledge bases and evaluation corpora. This is done by using mappings, lenses, and filters.

Mappings

Most gold standard corpora use DBpedia as a knowledge source. Nevertheless, with the recent success of other linked data initiatives such as Wikidata, GeoNames and linked government data, named entity linking approaches have started to draw upon these other sources that even may surpass DBpedia's quality for same tasks. To compare the performance of such approaches, a mapping between these linked data sources and the DBpedia gold standard annotations needs to be established. Orbis provides a possibility to map KB links by providing a new link for a specific link found in the gold standard.

Lenses

Gold standard corpora are created based on a specific KB version available at the time of creation but are often not updated or rebuilt when the KB changes and evolves. Due to this, annotator services running on newer versions of a KB are likely to find entities that have not been annotated in the original gold standard because these entities can only be found in recent KB versions and have not been previously available. Lenses provide means for adjusting to such circumstances by letting Orbis only consider predicted annotations of entities found in the desired KB version.

Filters

Gold standard corpora often follow different annotation styles and guidelines. For instance, a location entity such as "Washington, DC" might be either annotated as one entity (Washington, the capital of the United States), or as two (once as the city of Washington and once as the District of Columbia). Filters provide means toward mitigating such differences and ensuring that NEL components that follow different guidelines are treated fairly by allowing researchers to manually exclude certain annotations from the evaluation process.

Gold Standard Link Equivalence

Modifying a gold standard is not considered good practice, however, replacing the existing links from a gold with the equivalent links from another KB (e.g., replacing DBpedia links with Wikidata links) should be a good scenario, otherwise the loss of gold standards due to links from KBs that are not maintained anymore could become a common issue. Modern KBs have a convoluted history as most of them start life as a research project that is than taken over by a private company or association. Due to this process, some KBs are discontinued (e.g., Freebase), others changed owners (e.g., DBpedia), while others are continuously updated (e.g., Wikidata). This also means that some corpora annotated with older KBs (e.g., Freebase, early DBpedia versions <3.9) might not be usable today. In order to update them, a mechanism to update these links is needed and lenses do enable this mechanism. A direct translation of the links involves simply collecting the corresponding links from the new KB (e.g., owl:SameAs, skos:exactMatch, skos:closeMatch) or checking that the old links are still okay in the current KB version (e.g., when converting DBpedia 3.7 links to DBpedia 2018 links). Such a translation is possible directly by using monocle, but it should only be used if the results are high quality, for example if (1) the number of correct annotations from the original corpora is high; (2) the number of missing annotations from the original corpora is small; and (3) the quality of the link alignment is high (e.g., over 90% of the entities have good equivalent links). If one wants to improve upon the original entities, then only by using the published annotation rules of the respective corpora.

Add-ons

Orbis provides an add-on infrastructure to auto detect add-ons and execute them. Unlike plug-ins that plug into the Orbis pipeline and are used for evaluation purposes, Orbis add-ons perform tasks outside of the Orbis evaluation runs. Orbis add-ons provided to date are a YAML configuration builder, an annotation comparison tool to view two evaluation runs next to each other and a corpus downloader to download corpora and the respective gold standard. A gold standard quality checking add-on is under development and will be released this year.

Corpus Downloader

In order to use or test Orbis, a corpus is needed to run these tests on. Orbis provides a corpus loader that downloads and converts gold corpus files into an Orbis corpus structure. NIF is supported to be used as corpus loading format, accessing the data from the NIF dataset and storing it in Orbis. This Orbis corpus structure also allows researchers to quickly access parts of the corpus for further analysis and can be saved with the annotation results and metrics in a single file for future reference.

Gold Standard Quality Check

Compiling gold standards is a very time and labor intensive task. We, therefore, plan to develop a gold standard quality checker add-on that provides heuristics for verifying the correctness of gold standard annotations.

These heuristics draw upon natural language processing methods that provide additional metadata by performing dependency parsing (e.g., Stanford, Spacy or custom components), part-of-speech tagging (e.g., to identify verbs between proper nouns) and named entity recognition (NER, i.e., to determine the entity's type such as location, organization and person).

The Orbis gold standard quality checker will then use these metadata to validate whether the gold standard annotations correspond to the expected usage of annotation types within the sentence and whether the named entity type determined by the NER component matches the gold standard annotation's entity type.

The Gold standard quality checker, can not only help (1) in illuminating issues with the original annotations, but also be adapted to (2) identify missing annotations (e.g., if the NER component identified an entity that has not been annotated),.

Methods for validating corpus quality are easily implemented with Orbis and should be used to ensure the validity of evaluation settings, since gold standards are expensive and rarely updated. We plan to further extend the capabilities of the gold standard quality checker to include more complex mechanisms for verifying gold standard quality.

CONCLUSION AND OUTLOOK

Orbis has proven to be useful for our research. It serves as a tool for NEL researchers to understand issues with their systems, and does not intend to rival or replace existing evaluation tools like GERBIL or TAC-KBP but is intended as an additional tool focusing on evaluation analyses. The pipeline architecture allows third party developers to utilize and extend Orbis. Further efforts are being made to not only allow NEL evaluations to be run but also other evaluation types for other tasks e.g., slot filling, POS tagging, sentiment analysis or dependency parsing.

A first public version of Orbis was released on Github[5] and future releases will be published to the same project account. We hope to not only provide researchers with a useful tool but also gain new Orbis developers that implement their own modules for various stages in the pipeline and help us to advance Orbis to become an evaluation tool for various tasks in the field of natural language processing.

REFERENCES

Brasoveanu, A., Rizzo, G., Kuntschik, P., Weichselbraun, A., & Nixon, L. J. B. (2018). Framing named entity linking error types. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Hachey, B., Nothman, J., & Radford, W. (2014). Cheap and easy entity evaluation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (Vol. *2*, pp. 464–469).

Ji, H., Nothman, J., Dang, H. T., & Hub, S. I. (2016). Overview of TAC-KBP2016 tri-lingual EDL and its impact on end-to-end cold-start KBP. In *Proceedings of TAC*.

Odoni, F., Kuntschik, P., Braşoveanu, A. M. P., & Weichselbraun, A. (2018). On the importance of drill-down analysis for assessing gold standards and named entity linking performance. *Procedia Computer Science*, *137*, 33–42. https://doi.org/10.1016/j.procs.2018.09.004

Röder, M., Usbeck, R., & Ngonga Ngomo, A.-C. (2017). GERBIL – benchmarking named entity recognition and linking consistently. *Semantic Web* (Preprint, 1–21.

[5] https://github.com/orbis-eval/Orbis