Docforia: A Multilayer Document Model

Marcus Klang and Pierre Nugues

Department of computer science
Lund University, Lund
marcus.klang@cs.lu.se, pierre.nugues@cs.lth.se

Abstract

In this paper, we describe **Docforia**, a multilayer document model and application programming interface (API) to store formatting, lexical, syntactic, and semantic annotations on Wikipedia and other kinds of text and visualize them. While Wikipedia has become a major NLP resource, its scale and heterogeneity makes it relatively difficult to do experimentations on the whole corpus. These experimentations are rendered even more complex as, to the best of our knowledge, there is no available tool to visualize easily the results of a processing pipeline. We designed Docforia so that it can store millions of documents and billions of tokens, annotated using different processing tools, that themselves use multiple formats, and compatible with cluster computing frameworks such as Hadoop or Spark. The annotation output, either partial or complete, can then be shared more easily. To validate Docforia, we processed six language versions of Wikipedia: English, French, German, Spanish, Russian, and Swedish, up to semantic role labeling, depending on the NLP tools available for a given language. We stored the results in our document model and we created a visualization tool to inspect the annotation results. The Docforia API is available at https://github.com/marcusklang/docforia.

1. Introduction

Wikipedia is one of the largest freely available encyclopedic sources: It is comprehensive, multilingual, and continuously expanding. These unique properties makes it a popular resource now used in scores of NLP projects such as translation (Smith et al., 2010), semantic networks (Navigli and Ponzetto, 2010), named entity linking (Mihalcea and Csomai, 2007), information extraction, or question answering (Ferrucci, 2012).

Nonetheless, the Wikipedia size, where many language versions have now more that one million of articles makes it more difficult to handle than "classic" and older corpora such as the Penn treebank (Marcus et al., 1993). Processing the complete collection of Wikipedia articles, or a part of it, is a nontrivial task that requires dealing with multiple markup variants across the language versions, multiple tools and storage models. In addition, the application of a NLP pipeline to carry out the annotation (tokenization, POS tagging, dependency parsing, and so on) is a relatively costly operation that can take weeks on a single computer.

Docforia is a multilayer document model to store formatting, lexical, syntactic, and semantic annotations on Wikipedia and other kinds of text and visualize them. To deliver results in a reasonable time, Docforia is compatible with cluster programming frameworks such as Spark or Hadoop. Using the Langforia language processing pipelines (Klang and Nugues, 2016a), we processed six language versions of Wikipedia: English, French, German, Spanish, Russian, and Swedish, up to semantic role labeling, depending on the NLP tools available for a given language. We stored the results in the document model. We designed an interactive visualization tool, part of Langforia, so that a user can select languages, documents, and linguistic layers and examine the annotation output.

2. The Document Model

We created the Docforia multilayer document model library to store, query, and extract hypertextual information common to many NLP tasks such as part-of-speech tagging, coreference resolution, named entity recognition and linking, dependency parsing, semantic role labeling, etc., in a standalone package.

This model is intended for large and heterogenous collection of text, including Wikipedia. We designed it so that we could store the original markup, as well as the the results of the subsequent linguistic processing. The model consists of multiple layers, where each layer is dedicated to a specific type of annotation. It is nondestructive and preserves the original white spaces.

The annotations are encoded in the form of graph nodes, where a node represents a piece of data: A token, a sentence, a named entity, etc., delimited by ranges. These nodes are possibly connected by edges as in dependency graphs. The data structure used is similar to a property graph and Fig. 1 shows the conversion pipeline from the Wikimedia dumps to the abstract syntactic trees (AST) and Docforia layers.

In contrast to the UIMA project (Ferrucci and Lally, 2004), which also provides an infrastructure to store unstructured documents, the Docforia library emphasizes simplicity, portability, ease of integration, minimal dependencies, and efficiency.

3. Annotation

We convert Wikipedia HTML dumps into Docforia records using an annotation pipeline. The first step converts the HTML documents into DOM trees using jsoup[1]. The second step extracts the original page structure, text styles, links, lists, and tables. We then resolve the links to unique Wikidata identifiers. These steps are common to all the language editions we process.

Wikidata is central to the multilingual nature of Docforia. Wikidata is an entity database, which assigns unique identifiers across all the language editions of Wikipedia. The University of Gothenburg, for instance, has the unique id:

[1]http://jsoup.org/

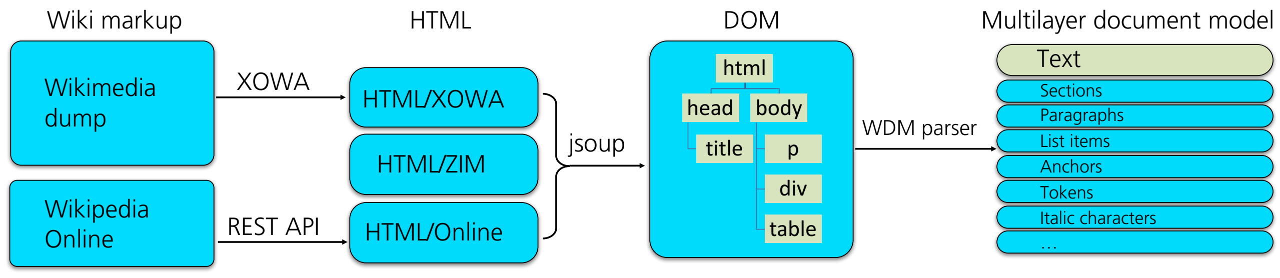


Figure 1: Conversion of Wikipedia dumps into abstract syntactic trees and the Docforia multilayer document model (Klang and Nugues, 2016b).

Q371522 that enables to retrieve the article pages in English, French, Swedish, or Russian.

In addition to the common processing steps and depending on the available tools, we can apply linguistic annotations that are language specific using Langforia. These annotations can range from a simple tokenization to semantic-role labels or coreference chains. We save all the results of the intermediate and final steps as files in the Parquet format. We selected this format as it is efficient and easy to use with the Apache Spark data processing engine.

4. Application Programming Interface

The Docforia API builds on the concepts of document storage and document engine. The document storage consists of properties, layers (node or edge layers) to store typed annotations, token, sentence, relationship, where the nodes can have a range, and finally sublayer variants: gold, predicted, coreference chains. The document engine defines query primitives such as covers, for instance the tokens in a anchor, transactions, and partial lightweight documents called views.

The Docforia data structure is similar to a typed property graph. It consists of nodes (tokens, sentences, paragraphs, anchors, ...), edges (connections between e.g tokens to form a dependency tree), and properties per node and edge (Token: pos, lemma, ...).

The piece of code below shows how to create tokens from a string and assign a property to a range of tokens, here a named entity with the *Location* label:

```
Document doc = new MemoryDocument(
  "Greetings from Lund, Sweden!");
// 0123456789012345678901234567 8

Token Greetings   =
  new Token(doc).setRange(0,  9);
Token from        =
  new Token(doc).setRange(10, 14);
Token Lund        =
  new Token(doc).setRange(15, 19);
Token comma       =
  new Token(doc).setRange(19, 20);
Token Sweden      =
  new Token(doc).setRange(21, 27);
Token exclamation =
  new Token(doc).setRange(27, 28);
```

```
Sentence greetingsSentence =
  new Sentence(doc).setRange(0, 28);

NamedEntity lundSwedenEntity =
  new NamedEntity(doc)
    .setRange(Lund.getStart(),
      Sweden.getEnd())
        .setLabel("Location");
```

The API provides SQL-like query capabilities and the code below shows how to find the named entities in a document:

```
NodeTVar<Token> T = Token.var();
NodeTVar<NamedEntity> NE =
  NamedEntity.var();

List<Token> lundLocation =
  doc.select(T, NE)
    .where(T).coveredBy(NE)
    .stream()
    .sorted(StreamUtils.orderBy(T))
    .map(StreamUtils.toNode(T))
    .collect(Collectors.toList());
```

5. Visualization

We built a front-end application, part of Langforia, to enable the users to visualize the content of Docforia-based corpora. This application has the form of a web server that embeds the Docforia library and Lucene to index the documents. We created a Javascript component for the text visualization on the client. This client provides a user interface for searching and visualizing Docforia data in the index. The layers are selectable from a dropdown menu and the supported visualizations are the ranges and relationships between them.

Figure 2 shows the annotations of the parts of speech, named entities, and dependency relations of the sentence:

> Göteborgs universitet är ett svenskt statligt universitet med åtta fakulteter, 37 000 studenter, varav 25 000 helårsstudenter och 6000 anställda.
>
> 'The University of Gothenburg is a Swedish public university with eight faculties, 37,000 students, (25,000 full-time), and 6,000 staff members.'

If we hover over the words, the visualizer (see Sect. 5.) shows the properties attached to a word in CoNLL-like format. In Fig. 3, the properties correspond to the word *Vasaparken*.

6. Conclusion and Future Work

We described Docforia, a multilayer document model, structured in the form of a graph. It enables a user to represent the results of large-scale multilingual annotations. Using it and the Langforia language processing pipelines, we annotated Wikipedia dump (Klang and Nugues, 2016a). When applied to Wikipedia, MLDM links the different versions through an extensive use of URI indices and Wikidata Q-numbers.

Together with Docforia, we used the Lucene library to index the records. The resulting system can run on a single laptop, even with multiple versions of Wikipedia.

Docforia is written in Java. In the future, we plan to develop a Python API, which will make it possible to combine Python and Java tools. This will enable the programmer to build prototypes more quickly as well as experiment more easily with machine learning algorithms.

Docforia is available from github at `https://github.com/marcusklang/docforia`.

Acknowledgements

This research was supported by Vetenskapsrådet, the Swedish research council, under the *Det digitaliserade samhället* program.

References

David Ferrucci and Adam Lally. 2004. UIMA: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348, September.

David Angelo Ferrucci. 2012. Introduction to “This is Watson”. *IBM Journal of Research and Development*, 56(3.4):1:1 –1:15, May-June.

Marcus Klang and Pierre Nugues. 2016a. Langforia: Language pipelines for annotating large collections of documents. In *Proceedings of Coling 2016, demonstration session*, Osaka, December. To appear.

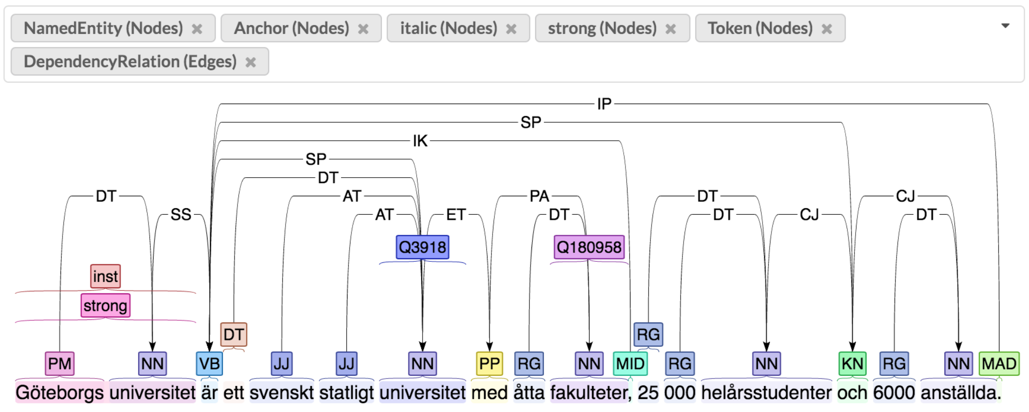
Marcus Klang and Pierre Nugues. 2016b. Wikiparq: A tabulated Wikipedia resource using the Parquet format. In *Proceedings of LREC 2016*, pages 4141–4148, Portorož, Slovenia.

Mitchell Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Rada Mihalcea and Andras Csomai. 2007. Wikify!: Linking documents to encyclopedic knowledge. In *Proceedings of the Sixteenth ACM Conference on CIKM*, CIKM ’07, pages 233–242, Lisbon, Portugal.

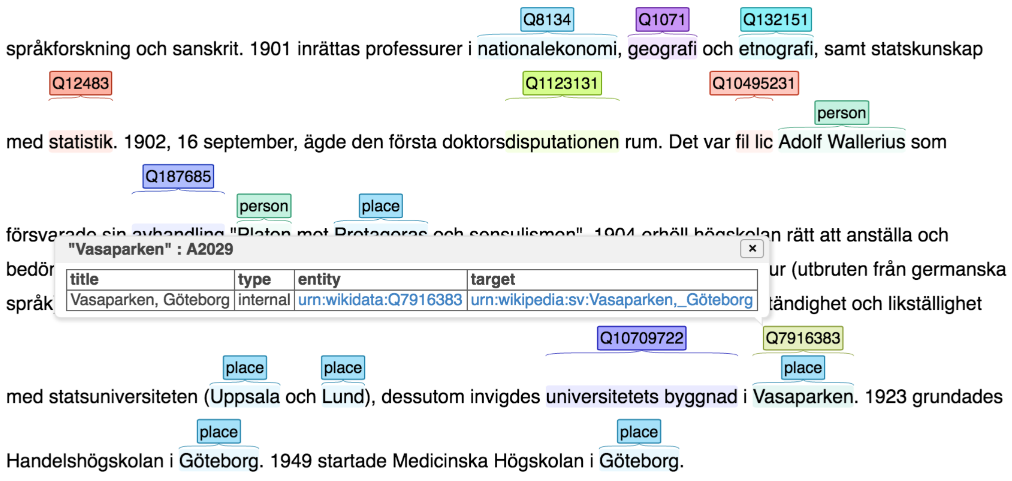
Roberto Navigli and Simone Paolo Ponzetto. 2010. Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the ACL*, pages 216–225, Uppsala.

Jason R. Smith, Chris Quirk, and Kristina Toutanova. 2010. Extracting parallel sentences from comparable corpora using document level alignment. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT ’10, pages 403–411.


NamedEntity (Nodes) ✖ Anchor (Nodes) ✖ italic (Nodes) ✖ strong (Nodes) ✖ Token (Nodes) ✖ DependencyRelation (Edges) ✖

Göteborgs universitet är ett svenskt statligt universitet med åtta fakulteter, 25 000 helårsstudenter och 6000 anställda.

Figure 2: Visualization of six layers including: Tokens, named entities, and dependency relations


språkforskning och sanskrit. 1901 inrättas professurer i nationalekonomi, geografi och etnografi, samt statskunskap
med statistik. 1902, 16 september, ägde den första doktorsdisputationen rum. Det var fil lic Adolf Wallerius som
försvarade sin avhandling "Platon mot Protagoras och sensualismen". 1904 erhöll högskolan rätt att anställa och
bedör ur (utbruten från germanska
språk tändighet och likställighet
med statsuniversiteten (Uppsala och Lund), dessutom invigdes universitetets byggnad i Vasaparken. 1923 grundades
Handelshögskolan i Göteborg. 1949 startade Medicinska Högskolan i Göteborg.

"Vasaparken" : A2029 ×

title	type	entity	target
Vasaparken, Göteborg	internal	urn:wikidata:Q7916383	urn:wikipedia:sv:Vasaparken,_Göteborg

Figure 3: Visualization of properties