

Interchange Formats for Visualization: LIF and MMIF

Kyeongmin Rim*, Kelley Lynch*, Marc Verhagen*, Nancy Ide[†], James Pustejovsky*

* Brandeis University, Department of Computer Science

{krim,kmlynch}@brandeis.edu, marc@cs.brandeis.edu, jamesp@brandeis.edu

[†] Vassar College, Department of Computer Science

ide@cs.vassar.edu

Abstract

Promoting interoperable computational linguistics (CL) and natural language processing (NLP) application platforms and interchangeable data formats have contributed improving discoverability and accessibility of the openly available NLP software. In this paper, we discuss the enhanced data visualization capabilities that are also enabled by inter-operating NLP pipelines and interchange formats. For adding openly available visualization tools and graphical annotation tools to the Language Applications Grid (LAPPS Grid) and Computational Linguistics Applications for Multimedia Services (CLAMS) toolboxes, we have developed interchange formats that can carry annotations and metadata for text and audiovisual source data. We describe those data formats and present case studies where we successfully adopt open-source visualization tools and combine them with CL tools.

Keywords: Interchange formats, Interoperability, Data visualization, Multi-media annotation

1. Introduction

In this paper, we discuss the enhanced data visualization capabilities enabled by the open-source natural language processing (NLP) tools (either separately or pipelined in application workflows) as well as openly available visualization toolkits. While popular text-based search and navigation applications such as ElasticSearch+Kibana stack (ELK)¹ or Solr² are beginning to provide some Human Language Technology (HLT)-enabled indexing that can then be visualized (e.g., named entity recognition (NER)), we argue that a more robust and generic capability for visualizing data is enabled by adopting common rich and flexible interchange formats from which the visualization can be mapped. To this end, we describe the use of the LAPPS Interchange Format (LIF) and the Multi-Media Interchange Format (MMIF) as pivots from which visualizations can be driven.

Specifically, we discuss two different scenarios for the visualization of data: (i) visualization of an *individual* document and its annotations; and (ii) visualization of a *collection* of documents that is processed offline in order to ultimately provide a display of collection-wide phenomena. We demonstrate how visualization of an individual document and its annotations is performed by converting between pivoting interchange formats (specifically LIF and/or MMIF) and application specific formats. Using interchange formats enables the chaining of tools into a pipeline which generates automatic annotations that can then be visualized or modified by a human-in-the-loop in multiple annotation environments. Additionally, we describe the functionality enabled by using LIF and MMIF as interchange formats within the Language Applications Grid (LAPPS Grid)-Galaxy and Computational Linguistics Applications for Multimedia Services (CLAMS) respectively. The second scenario involves the visualization of data characteristics that inhere over an entire collection. We illustrate how MMIF and LIF can facilitate visualization across a collec-

tion of documents by being exported for indexing in document indexing tools (ELK or Solr) and associated front-end web interfaces (such as Kibana) that support per-index visualization .

2. Interchange Formats and Interoperability

Since we use interchange formats as the mediators between data and their visualizations we describe here the two formats that we have used in our work.

2.1. LAPPS Interchange Format

As a data driven research community, it has been one of most important goals shared among the computational linguistics (CL) community to have a common data format that can be used in different data processing projects. Within the CL community, the Unstructured Information Management Applications (UIMA) framework (Ferrucci et al., 2009) and the General Architecture for Text Engineering (GATE) (Cunningham et al., 2013) have been served as well-established and popular tool-chaining platforms for researchers and NLP developers. Although GATE focuses primarily on textual data, UIMA provides an extremely general model of type systems and annotations that can be applied to multimedia source data. However, there is a steep learning curve supporting UIMA’s generality, due in large part to its tight binding to XML syntax and the Java programming language. More recently, web-based workflow engines such as the LAPPS Grid (Ide et al., 2014) and WebLicht (Hinrichs et al., 2010) have been developed that provide user-friendly web interfaces for chaining NLP tools. These platforms not only offer tool repositories containing state-of-the-art NLP tools for annotating textual data at a variety of linguistic levels (e.g., CoreNLP (Manning et al., 2014), OpenNLP (OpenNLP, 2017), UDPipe (Straka and Straková, 2017)), but also provide open source software development kits (SDKs) for tool developers in order to promote adoption. The LAPPS Grid and WebLicht both provide for chaining tools from different developers, which use a variety of I/O formats, by virtue of underlying data interchange formats that impose a common I/O format

¹<https://www.elastic.co/>

²<https://lucene.apache.org/solr/>



Figure 1: Named entity annotations in the LAPPS Grid, generated by LINDAT/CLARIN’s NameTag⁴ tool and visualized with Brat.

among those tools. The LAPPS Grid uses LIF (Verhagen et al., 2015), a JSON-LD serialization, as its interchange format; while WebLicht uses its XML-based Text Corpus Format (TCF) (Heid et al., 2010). Additionally, the LAPPS Grid defines a linked data vocabulary that ensures semantic interoperability (Ide et al., 2015). Beyond in-platform interoperability, the LAPPS Grid has established multi-platform interoperability between LAPPS Grid and two CLARIN platforms (Hinrichs et al., 2018) as well as several other platforms (e.g., DKPro (Eckart de Castilho and Gurevych, 2014), PubAnnotation (Kim and Wang, 2012), and INCEPTION (Klie et al., 2018)).

Figure 1 shows a visualization of named entity annotations in the LAPPS Grid, using Brat (Stenetorp et al., 2012). All annotations are represented in the LIF format and linked either to offsets within read-only primary data or to other annotation layers. Within the LIF document containing the annotations, each annotation references a name (e.g., PERSON), possibly coupled with additional attributes, that links to a full definition in the LAPPS Grid Web Service Exchange Vocabulary (WSEV)⁵. Alternative names used within specific tools are mapped to the WSEV in order to ensure semantic consistency among tools from different sources.

Although it is not shown in the trimmed-down screenshot in fig.1, the `views` list contains a view generated by NameTag and that view includes a list of annotation where one looks like the fragment below.

```
{
  "@type": "NamedEntity",
```

⁵<https://vocab.lappsgrid.org/>. Note the reference in the `@context` field of the LIF tool output.

```
"id": "c0",
"start": 24,
"end": 38,
"features": {
  "category": "PERSON" }
}
```

The `@type` attribute’s value is a shorthand for the full form `http://vocab.lappsgrid.org/NamedEntity` which contains the definition of Named Entity annotation type. The annotation is anchored in the text, and its feature dictionary gives the information relevant for named entity annotations. All annotations in LIF follow this format.

2.2. Multi-Media Interchange Format

Recent developments driven by advancements in high data throughput and machine learning algorithms have brought impressive boosts in performance of not only NLP, but also computer vision (CV), and speech technologies processing audio and video data. The machine learning approach to solving problems is data-driven, and most state-of-the-art applications are based on supervised algorithms which rely on large sets of training data. To ensure the high quality of datasets containing rich multimodal annotations, the CL community has had to move beyond text-only annotation practices, and has tried to establish a common format for multimodal annotations, particularly with regard to annotating speech in audio and gestures in video. For example, (Schmidt et al., 2008) outline a diversity of annotation applications and formats, as well as the community effort to develop an interoperable format that carries complicated, layered multimodal annotation. As a result,

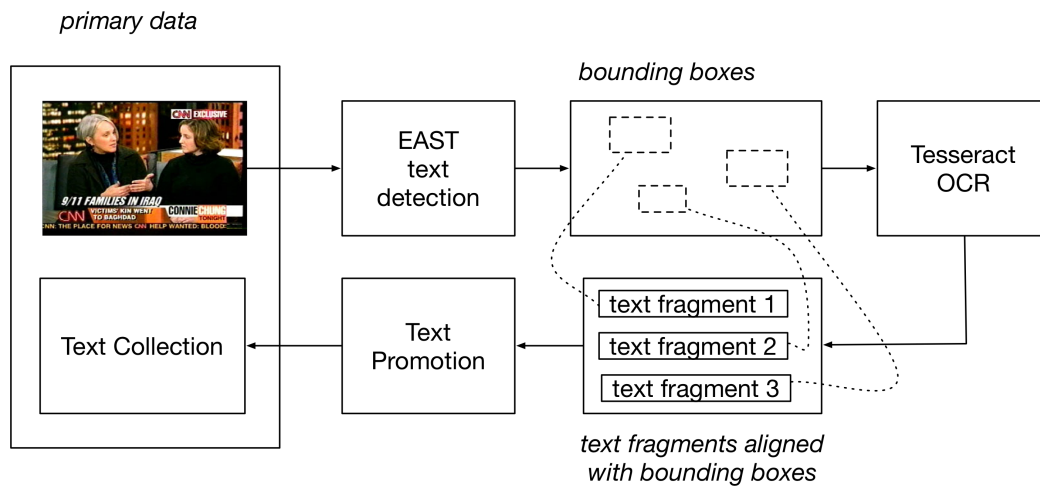


Figure 2: A primary data text collection can be created from an image and can then be input to downstream NLP components.

UIMA and Component MetaData Infrastructure (CMDI) (Broeder et al., 2012) have been widely adopted frameworks that provide interoperable multimodal information exchange between computational analysis tools, and for metadata repositories for discoverability, respectively. More recently, the International Image Interoperability Framework (IIIF)⁶ has been gaining popularity among the Libraries, Archives and Museums (LAM) community. IIIF is an industry-led project, which originally started with the goal of providing an interchange format to transfer collections of born-digital or digitized images (e.g., scanned books, sheet music, pictures, and paintings) along with textual annotation and metadata on items in the collections. In its latest development, starting from its API version 3.0 (in *beta* stage as of time of writing)⁷, IIIF supports not only images, but also time-based audiovisual data. The main purpose of this specification is to provide consumers of these collections (usually digital libraries, archives, and museums) with a consistent semantics of how to present the collections in their client software (e.g., in what order, in which orientation, on what zoom level, etc). Unfortunately, however, the IIIF does not provide detailed specifications for the semantics of the content of the textual annotations. In principle, one could design an independent, adequate data model for textual annotation that can be carried out on IIIF, since the IIF specification is built upon the Open Annotation model and linked-data conformity, but this would involve significant additional effort.

MMIF is an interoperable representation format that is used in the CLAMS (Rim et al., 2019) project. CLAMS is a platform of computational analysis tools designed for digital libraries and archives who have to deal with not just textual data, but also audiovisual time-based data. To handle the complexity of multimodal content and semantics of the audiovisual data sources, MMIF is specifically designed to enable alignment of annotations on different modes of the primary data sources.

Specifically, multimodal annotations in MMIF are first categorized by the anchor type on which the annotation is placed. That is, an annotation can be placed on: (1) character offsets of a text; (2) time segments of time-based media; (3) two-dimensional ($w \times h$) or three-dimensional ($w \times h \times \text{duration}$) bounding boxes on video frames; and (4) other annotations. For instance, a NER annotation can anchor on a `token` annotation that is in turn anchored on character offsets. Furthermore, the characters can be from primary text data or from other annotations (such as automatic speech recognition (ASR) or optical character recognition (OCR)).

MMIF is also built upon the same philosophy as LIF in representing annotations, in that it distinguishes between the primary data and annotation layers (called `views`) on those data. The primary data in LIF is the text and it is made available as a read-only entity that annotations can refer to. For MMIF there is not a single text that comprises the primary data but a set of media: images, videos, audio streams and texts. While texts can be represented directly in the MMIF object, other media are typically referred to by URL or local file path because of the size of data. MMIF also allows a bit more structure in its primary data, in that its individual media can be collections rather than just single instances; an example of where this is used will follow later in this section. The annotation elements in an MMIF view are very similar to the ones in LIF, except that there is a wider range of anchors. Where LIF annotation can only refer to text offsets or other annotations, MMIF annotation can anchor into image, audio and video sources as well. Another difference from LIF is that media can be associated with an alignment: for example, an audio stream can be associated with a transcription.

Figure 2 shows how MMIF and CLAMS allow text processing to occur in a multimedia workflow. We start with an image and run that image through the EAST text detection tool (Zhou et al., 2017). In MMIF, the results are stored as annotations where each annotation is anchored to a bounding box in the image. These bounding boxes are

⁶<https://iiif.io/>

⁷<https://iiif.io/api/presentation/3.0/>

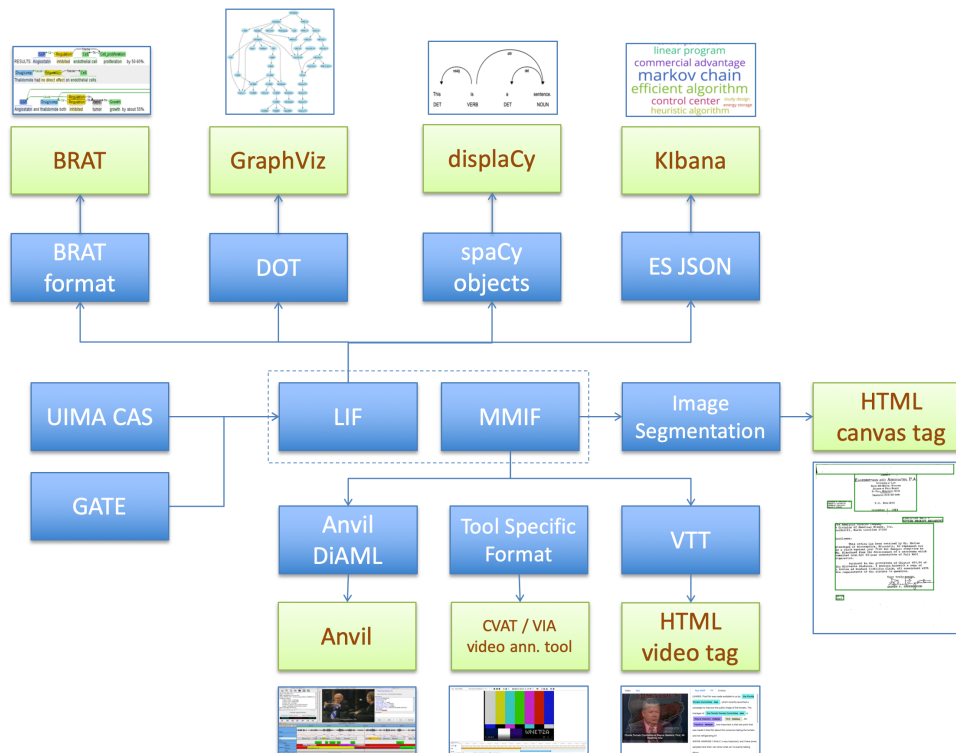


Figure 3: LIF and MMIF act as pivots and mediate between a variety of data and visualizations.

then handed to the Tesseract OCR tool⁸ which generates text fragments which are linked in MMIF to the bounding box annotations generated by EAST. At that point we run into a problem if we wish to apply some language processing on those fragments. It is rather cumbersome for an NLP tool to find these fragments, since they are expressed as annotations themselves. To deal with this, we write specialized modules that take texts generated in a variety of scenarios and promote that text to a collection of primary texts. This collection is obviously derived from an image (and its metadata specify that) and therefore perhaps it is not technically primary data, but from the perspective of downstream text processing, it is convenient to view the generated data as primary data that cannot be amended.

3. Case Studies

This section presents some illustrative case studies where the MMIF and LIF interchange formats have been used to mediate between data and visualizations associated with this data.

3.1. LIF: GraphViz and Brat

The Galaxy-based instances of the LAPPS Grid include plugins for visualization of annotations over individual documents that can be invoked by the user at any step in an annotation workflow or following the application of an individual annotation engine. Brat visualization includes *text-bound* annotations, expressed in Brat’s internal flat-file standoff format via character offsets; and directional, binary *relation* annotations between text-bound annotations

consisting of one or more contiguous tokens. Mapping between LIF and Brat’s standoff format is relatively straightforward, although without collapsing two or more LIF “views” (typically, one annotation type) into a single view, we are constrained to display only one annotation layer at a time. Mapping LIF to GraphViz’s⁹ flat-file “DOT” format is similarly straightforward. Additionally, mapping from LIF to the internal formats used in the PubAnnotation and INCEption platforms allows for seamless import and export of LIF documents to and from their single-document annotation visualizer/editors.

3.2. MMIF: displaCy

As a part of development of CLAMS and MMIF, we developed a standalone Python-based webapp for visualizing different annotations from a given MMIF json input. Using CLAMS python-SDK, we were able to directly map MMIF json objects to python native objects and use spaCy and displaCy (Honnibal and Montani, 2017) to visualize linguistic annotations (e.g. NER, `trees`) for rendering in web browsers.

Using MMIF it is possible to apply NLP tools to text generated with CV or OCR applications. One possible workflow allows the extraction and processing of lower-thirds (graphic overlay placed in the lower area of the videos, for on-screen subtitles, captions, personal identities) text in a news broadcast video for visualization in displaCy. We build upon the multimedia workflow described in 2.2. First, text is localized and recognized using the EAST and Tesseract tools respectively. Using the location and duration of

⁸<https://github.com/tesseract-ocr>

⁹<https://graphviz.gitlab.io/about/>



Figure 4: Video time segments generated in CLAMS imported into VIA for manual annotation.

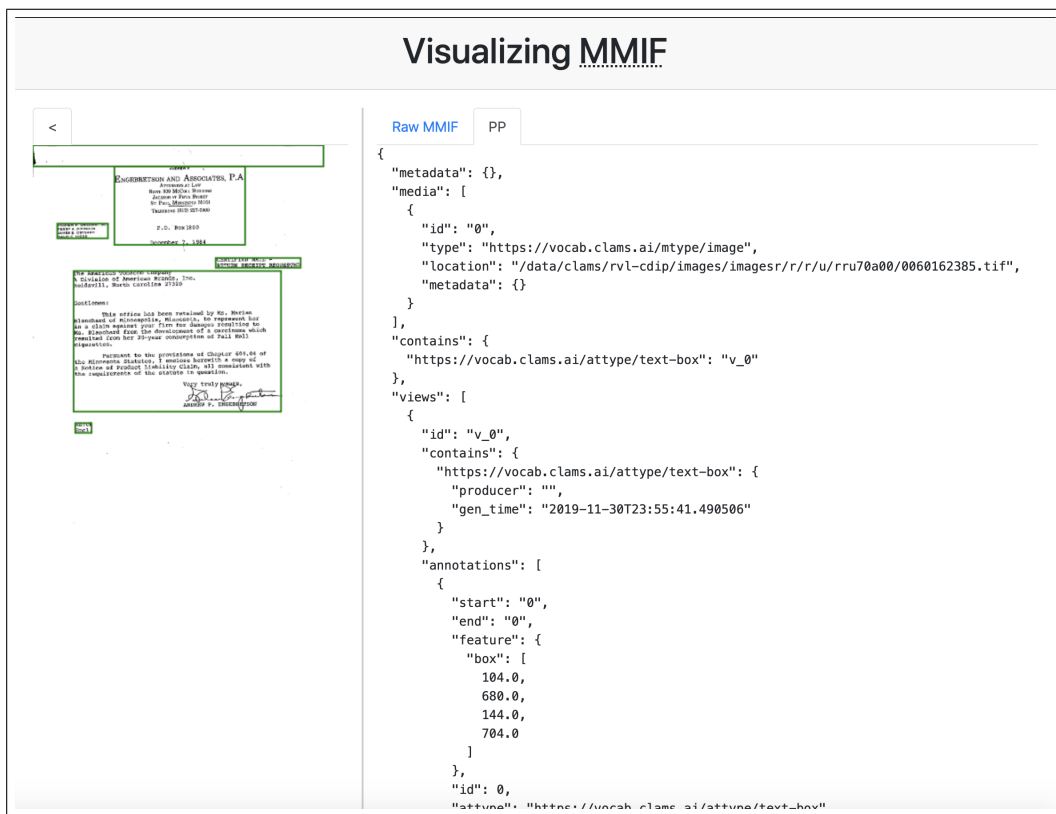


Figure 5: Visualizing bounding boxes generated via a RLSA tool on a document from the RVL-CDIP dataset. Visualization is on the left, while source MMIF annotation is on the right.

the text region, we can identify text that it likely to be a lower-third entry. This text can be promoted to primary data as described in 2.2., where it can be processed by NLP tools. Finally, the linguistic annotations can be visualized with displaCy. Additionally, displaCy can be used to display annotations on the transcript of a video as shown in Figure 6.

3.3. MMIF: Bounding Boxes in Images

Document segmentation is an important step in processing digitized documents. The bounding boxes that result from performing document segmentation can be visualized in the MMIF visualizer web application through the HTML5 <canvas> tag. RLSA (Wong et al., 1982) is one algorithm for document segmentation. In Figure 5, bounding boxes' coordinates have been generated by running a tool written with the CLAMS SDK and run through the CLAMS Galaxy interface. The resulting MMIF is displayed through

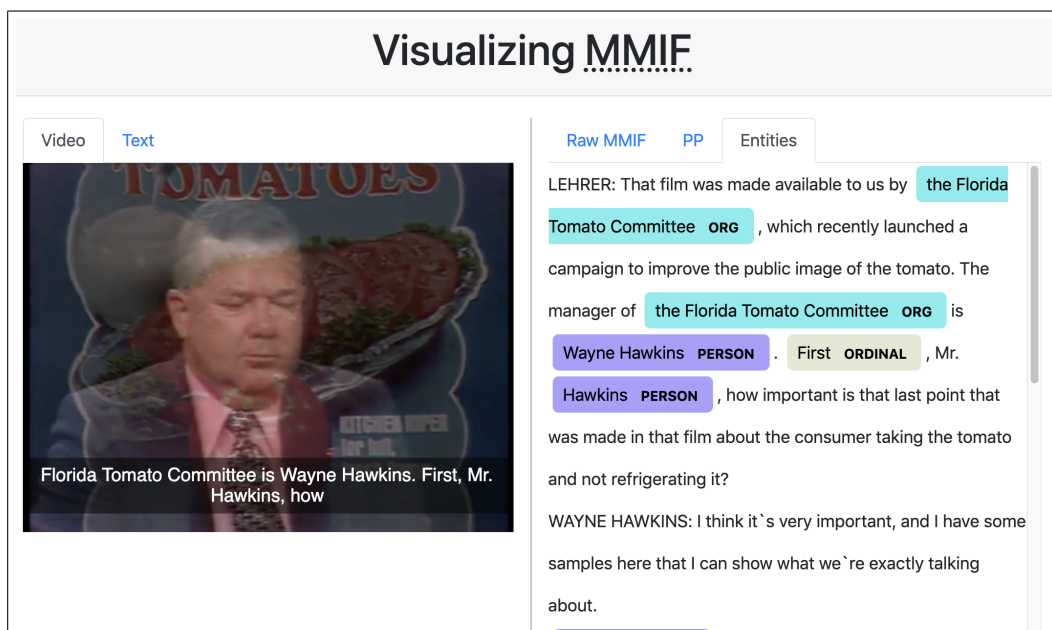


Figure 6: Force-aligned transcript is then processed for synchronized video caption (exported to WebVTT format) as well as for named entity visualization (using dispalyCy). Results are shown in a standalone web application.

the MMIF visualization application.

3.4. MMIF: Synchronized Captions for Web Video Player

Forced alignment annotations allow an existing transcript to be time-aligned to the audio stream. The time aligned annotations can be stored in the view of an MMIF then exported to a *standard* WebVTT¹⁰ file which can be used to provide synchronized captions to a video displayed on a web page using HTML5 `<video>` and `<track>` tags. MMIF facilitates the generation of time-aligned transcripts by providing a shared vocabulary for chaining tools into a workflow. The performance of existing forced alignment tools such as the Gentle forced aligner (Ochshorn and Hawkins, 2015) and the Montreal forced aligner (McAuliffe et al., 2017) declines with audio that contain non-speech segments. In the American Archive of Public Broadcasting (AAPB), news broadcasts often contain segments of music or commercials. In order to apply forced alignment tools to these types of media, it is necessary to filter out any significant portions of non-speech audio. The CLAMS platform enables filtering non-speech segments of video with various tools such as SMPTE Bar segment identification. A segment of time-aligned transcript is shown as a caption on the video in the left panel of Figure 6.

3.5. MMIF: CVAT and VIA

Various annotation tools and formats exist for video and image annotation. Two such tools are VIA (Dutta and Zisserman, 2019) and CVAT (OpenCV, 2018). These tools provide different but overlapping functionality. CVAT provides functionality for interpolating bounding box locations between frames. VIA provides a convenient interface for time segment annotation. Both of these annotation tools

enable a user to annotate images or videos with labeled bounding boxes or polygons. Additionally, CVAT supports polyline and keypoint annotations. CVAT also allows for pre-annotation within the platform through integration with the OpenVino toolkit¹¹. Each of these annotation tools allow a user to import existing annotations, however they require formats specific to the tool. CVAT supports importing and exporting annotations in multiple formats. VIA supports importing annotations from a VIA project or as a csv. MMIF can serve as a pivot between VIA, CVAT, and other annotation tools. We can convert MMIF to the CVAT XML v1.1 format to load annotations into the CVAT application. From the CVAT application, annotations can be exported to the CVAT XML 1.1 format which can then be mapped to MMIF. Various types of pre-annotations can be generated through the use of tools within the CLAMS platform. One example workflow is for annotation of text within lower thirds of a broadcast news video. First, shot changes are detected using the wrapped pySceneDetect¹² tool within CLAMS. Next, portions of the video containing 'junk frames' such as SMPTE bars or all black frames are identified (as shown in figure 4). These annotations can then be exported from MMIF to the VIA project format which is also json. By detecting shots and junk frame segments in advance, an annotator can more quickly annotate each shot for the presence of lower thirds and transcribe the contents of the lower thirds. Annotations can be converted from the VIA project format to MMIF.

3.6. LIF: Collection Visualization via Kibana

The previous sections discussed visualizations that display an individual document and its annotations. In addition to these types of visualizations, MMIF and LIF can facilitate

¹⁰<https://www.w3.org/TR/webvtt1/>

¹¹<https://software.intel.com/en-us/openvino-toolkit>

¹²<https://pyscenedetect.readthedocs.io/>

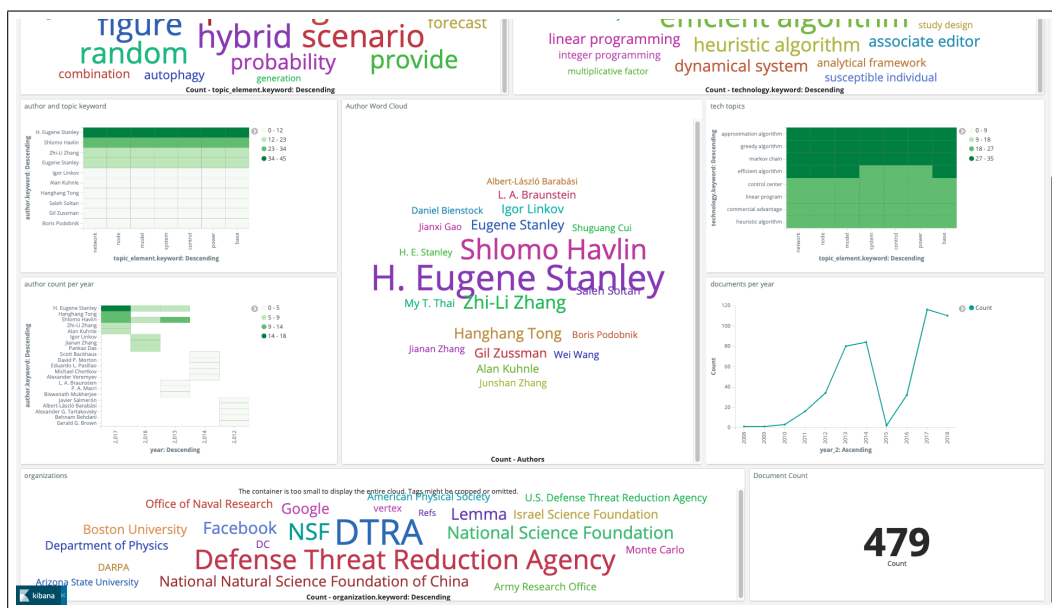


Figure 7: The Kibana Dashboard shows collection visualization as word clouds and conditional frequency distributions of named entities and topic keywords extracted using NLP pipelines enabled by LAPPS Grid.

visualization across a collection of documents by being exported for indexing in ELK or Solr. We have created a LAPPS workflow to extract a variety of information from a set of scientific articles where the information extracted included metadata (title and author), technology terms and topics. In addition we created a component that takes the information from the views and generates JSON documents for ELK indexing. This component can be easily extended to take in other types of annotations in LIF format. With the resulting document indices, we can then generate visualizations of the data set as a collection using Kibana’s built-in tools, and customizable JavaScript plugins, see Figure 7.

4. Conclusion and Future Directions

In this paper, we have presented an architecture for NLP-enabled data visualization through the use of interchangeable formats, in particular LIF and MMIF. We have demonstrated that the syntactic and semantic interoperability inherent in both MMIF and LIF facilitates not only the interoperability of multimodal analysis and annotation tools, but also a variety of data visualizations on the annotations created by these tools.

We are currently working on expanding capability of MMIF to make it fully compatible with LIF. With this full compatibility, our goal is to adopt text processing capability of LAPPS Grid platform into the CLAMS and to promote accessibility to CL applications not only within the CL community but also in LAM and Digital Humanity (DH) communities by providing with easy-to-use multimedia analysis toolkits that can help researchers using visual and audiovisual historical material.

Acknowledgements

We would like to thank the reviewers for their helpful comments. This work was supported by a grant from the National Science Foundation to Brandeis University and Vas-sar University, and by a grant from the Andrew W. Mellon

Foundation to Brandeis University. The points of view expressed herein are solely those of the authors and do not represent the views of the NSF or the Andrew W. Mellon Foundation. Any errors or omissions are, of course, the responsibility of the authors.

References

Broeder, D., Windhouwer, M., Van Uytvanck, D., Goosen, T., and Trippel, T. (2012). CMDI: a component metadata infrastructure. In *Describing LRs with metadata: towards flexibility and interoperability in the documentation of LR workshop programme*.

Cunningham, H., Tablan, V., Roberts, A., and Bontcheva, K. (2013). Getting more out of biomedical documents with GATE’s full lifecycle open source text analytics. *PLoS computational biology*, 9(2):e1002854.

Dutta, A. and Zisserman, A. (2019). The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia, MM ’19*, New York, NY, USA. ACM.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Ferrucci, D., Lally, A., Verspoor, K., and Nyberg, E. (2009). Unstructured information management architecture (UIMA) version 1.0. OASIS Standard.

Heid, U., Schmid, H., Eckart, K., and Hinrichs, E. (2010). A Corpus Representation Format for Linguistic Web Services: The D-SPIN Text Corpus Format and its Relationship with ISO Standards. In *LREC2010*, Valletta, Malta, May. European Language Resources Association (ELRA).

- Hinrichs, E., Hinrichs, M., and Zastrow, T. (2010). Weblight: Web-based LRT services for German. In *Proceedings of the ACL 2010 System Demonstrations*, pages 25–29. Association for Computational Linguistics.
- Hinrichs, E., Ide, N., Pustejovsky, J., Hajic, J., Hinrichs, M., Elahi, M. F., Suderman, K., Verhagen, M., Rim, K., Stranak, P., and Misutka, J. (2018). Bridging the LAPPS Grid and CLARIN. In *LREC2018*, Miyazaki, Japan, May. European Language Resources Association (ELRA).
- Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.
- Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., Wang, D., Suderman, K., Verhagen, M., and Wright, J. (2014). The language application grid. In *LREC2014*, Reykjavik, Iceland, May. European Language Resources Association (ELRA).
- Ide, N., Suderman, K., Verhagen, M., and Pustejovsky, J. (2015). The language application grid web service exchange vocabulary. In *International Workshop on Worldwide Language Service Infrastructure*, pages 18–32. Springer.
- Kim, J.-D. and Wang, Y. (2012). PubAnnotation: A Persistent and Sharable Corpus and Annotation Repository. In *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing, BioNLP '12*, pages 202–205, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Klie, J.-C., Bugert, M., Boullosa, B., de Castilho, R. E., and Gurevych, I. (2018). The INCEption platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9. Association for Computational Linguistics.
- Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.
- McAuliffe, M., Socolof, M., Mihuc, S., Wagner, M., and Sonderegger, M. (2017). Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi. In *InterSpeech*, pages 498–502.
- Ochshorn, R. M. and Hawkins, M. (2015). Gentle forced aligner. <https://lowerquality.com/gentle/>. Accessed: 2019-11-08.
- OpenCV. (2018). Computer vision annotation tool (CVAT). <https://github.com/opencv/cvat>. Accessed: 2020-02-20.
- OpenNLP. (2017). Apache OpenNLP. <https://opennlp.apache.org/>. Accessed: 2020-02-20.
- Rim, K., Lynch, K., and Pustejovsky, J. (2019). Computational Linguistics Applications for Multimedia Services. In *Proceedings of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 91–97, Minneapolis, USA. Association for Computational Linguistics.
- Schmidt, T., Duncan, S., Ehmer, O., Hoyt, J., Kipp, M., Loehr, D., Magnusson, M., Rose, T., and Sloetjes, H. (2008). An exchange format for multimodal annotations. In *International LREC Workshop on Multimodal Corpora*, pages 207–221. Springer.
- Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: A Web-based Tool for NLP-assisted Text Annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, EACL '12*, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Straka, M. and Straková, J. (2017). Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.
- Verhagen, M., Suderman, K., Wang, D., Ide, N., Shi, C., Wright, J., and Pustejovsky, J. (2015). The lapps interchange format. In *International Workshop on Worldwide Language Service Infrastructure*, pages 33–47. Springer.
- Wong, K. Y., Casey, R. G., and Wahl, F. M. (1982). Document analysis system. *IBM Journal of Research and Development*, 26(6):647–656, November.
- Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., and Liang, J. (2017). EAST: an efficient and accurate scene text detector. *CoRR*, abs/1704.03155.