Towards Faster Annotation Interfaces for Learning to Filter in Information Extraction and Search

Carlos A. Aguirre
Dept. of Computer Science
caguirre97@ksu.edu

Shelby Coen
Dept. of Electrical and
Computer Engineering
shelby88@ksu.edu

Maria F. De La Torre
Dept. of Computer Science
marifer2097@ksu.edu

William H. Hsu
Dept. of Computer Science
bhsu@ksu.edu

Margaret Rys
Department of Industrial and Manufacturing
Systems Engineering
malrys@ksu.edu

Kansas State University
Manhattan, KS, United States

ABSTRACT

This work explores the design of an annotation interface for a document filtering system based on supervised and semi-supervised machine learning, focusing on usability improvements to the user interface to improve the efficiency of annotation without loss of precision, recall, and accuracy. Our objective is to create an automated pipeline for information extraction (IE) and exploratory search for which the learning filter serves as an intake mechanism. The purpose of this IE and search system is ultimately to help users create structured recipes for nanomaterial synthesis from scientific documents crawled from the web. A key part of each text corpus used to train our learning classifiers is a set of thousands of documents that are hand-labeled for relevance to nanomaterials search criteria of interest. This annotation process becomes expensive as the text corpus is expanded through focused web crawling over open-access documents and the addition of new publisher collections. To speed up annotation, we present a user interface that facilitates and optimizes the interactive steps of document presentation, inspection, and labeling. We aim towards transfer of these improvements to usability and response time for this annotator to other classification learning domains for text documents and beyond.

Author Keywords

annotation, document categorization, human-computer interaction, information retrieval, information extraction, machine learning

ACM Classification Keywords

Information systems → Information retrieval → **Users and interactive retrieva**l; Retrieval tasks and goals → **question answering, document filtering, information extraction**; machine learning → supervised learning **supervised learning by classification**

ESIDA'18, March 11, Tokyo, Japan.

INTRODUCTION

This paper addresses the task of *learning to filter* [7] for information extraction and search, specifically by developing a user interface for human annotation of documents. These documents are in turn used to train a machine learning system to filter documents by conformance to pre-specified formats and topical criteria. The purpose of filtering in our extraction task context centers around *question answering (QA)*, a problem in information retrieval (IR), information extraction (IE), and natural language processing (NLP) that involves formulating structured responses to free text queries. Filtering for QA tasks entails restricting the set of source documents, from which answers to specific queries are to be extracted.

Our overarching goal is to make manual annotation more affordable for researchers, by reducing the annotation time. This leads to the technical objectives of optimizing the presentation, interactive viewing, and manual annotation of objects without loss of precision, accuracy, or recall. This annotation is useful in many scientific and technical fields where users seek a comprehensive repository of publications, or where large document corpora are being compiled. In these fields, machine learning is applied to select and prepare data for various applications of artificial intelligence, from cognitive services such as question answering, to document categorization. Ultimately, annotation is needed not only to deal with the *cold start problem* [10] of personalizing a recommender system or learning filter, but also to keep previous work up to date with new document corpora [4]. Manual annotation is expensive because it requires expertise in the topic and because of the time taken in the process.

Currently, there are fields such as materials science and bioinformatics where annotation is needed to produce ground truth for learning to filter [2]. For this we have created a lightweight PDF annotation tool to classify documents based on relevance.

This annotation tool was developed with the goal to be more efficient and accurate than normal document annotation. The task is to filter documents based on content relevance, potentially reducing the size of the result set returned in response to a search query. This can boost the precision of search while also supporting information extraction for data mining by returning selected documents that are likely to contain domain-specific information, such as recipes for synthesizing a material of interest [6]. This can include passages and snippets recipes to be extracted for the synthesis of materials of interest. Analogous to this is annotating documents by category tagging. In this paper, classification is used to determine the **eligibility (by format)** and **relevance** of a candidate document, and annotation refers to the process of determining both eligibility and relevance. The purpose of this paper is to record and test this annotation tool with a relatively large subject group.

Background

In recent years, the growth of available electronic information has increased the need for text mining to enable users to extract, filter, classify and rank relevant structured and semi structured data from the web. Document classification is crucial for information retrieval of existing literature. Machine learning models based on global word statistics such as TF-IDF, linear classifiers, and bag-of-words support vector machine classifiers, have shown remarkable efficiency at document classification. The broad goal of our research is to extract figures and instructions from domain-specific scientific publications to create organized recipes for nanomaterial synthesis, including raw ingredients, quantity proportions, manufacturing plans, and timing. This task involves classification and filtering of documents crawled from the web.

The filtering task is framed in terms of topics of interest, specifically a dyad (pair) consisting of a known material and morphology. This in turn supports question answering (QA) tasks defined over **documents that are about this query pair**. For example, a nanomaterials researcher may wish to know the effective concentration and temperature of surfactants and other catalysts, to achieve a chemical synthesis reaction for producing a desired nanomaterial. [6]

Collecting information about a document's representation involves syntactic and semantic attributes, domain ontology and tokenization. Through the process of linguistically parsing sentences and paragraphs, semantic analysis extracts key concepts and words relevant to the aimed domain topic that are then compared to the taxonomy. In our work, this extraction involves inference and supervised learning to determine different sections using metadata attributes such as font, text-size and spatial location, along with natural language processing. Data and knowledge retrieval is dependent on finding documents that contain information about the synthesis of nanomaterials. Our approach is to use annotation-based learning, along with TF-IDF and a bag-of-words classifier to obtain relevant documents. This approach requires tagging and manual classification of documents to train the classifier-learning algorithm.

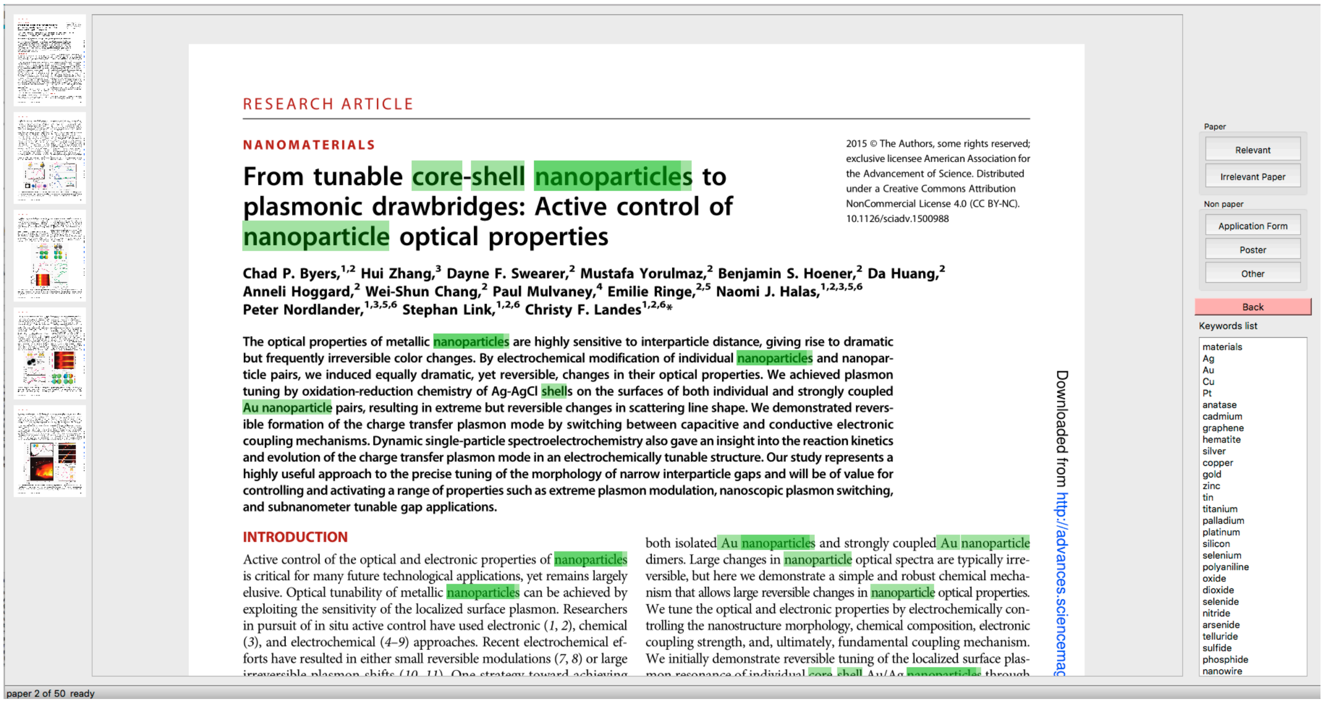
The document corpora that the paper focuses on is in the area of chemistry in synthesis of nanomaterial. We have constructed a custom web crawler to retrieve and filter documents in this area of research. The filtering process checks for the presence of a *gazetteer* – a list of words in the documents (TF-IDF) as best described in [1]. Gazetteers in information extraction (IE) are so named as generalizations of the geographical dictionaries used in maps and atlases. This process is only intended to filter documents based on the vocabulary. On the other hand, other criteria might be needed to determine the relevance of the documents. Because metadata in these documents is not always available, a learning to filter algorithm is necessary.

Need for a Fast Annotator

While many search engines provide a mechanism for explicit relevance feedback, past work on rapid annotation has mostly focused on markup for chunk parsing and other natural language-based tasks. For example, the Basic Rapid Annotation Tool (BRAT) of Stenetorp *et al.* [11] is designed to provide assistance in marking up entities and relationships at the phrase level. Meanwhile, fast annotators designed for information extraction (IE) are often focused on a knowledge capture task that is ontology-informed, such as in the case of the *Melita* framework for Ciravegna *et al.* [3] and the MMAX tool of Müller and Strube [8].

We seek to produce a reconfigurable tool for explicit relevance feedback for learning to filter that can make use of not only text features, but also domain-specific features (such as named entities detected using a gazetteer) and metadata features (such as formatting for sidebars, equations, graphs, photographs, other figures, and procedures). The longer-term goal for intelligent user interface design is to incorporate **user-specific** cues for relevance determination. These include actions logged from the user interface such as scrolling and searching within the document, but may be extensible to gaze tracking data such as scan paths and eye fixations. [5]

Manual annotation of training data brings a high cost in time due to the amount of training examples needed. Challenges in human annotation extend from time consumption to inconsistency in labeled data. The variety in the annotators' domain expertise among other human factors can create inaccurate and problematic training data. In the present work, an annotator user interface was developed to optimize the human annotation process by providing previews of document pages and highlighting relevant keywords. The increase in speed, user interface design and annotator biases



RESEARCH ARTICLE

NANOMATERIALS

From tunable core-shell nanoparticles to plasmonic drawbridges: Active control of nanoparticle optical properties

2015 © The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. Distributed under a Creative Commons Attribution NonCommercial License 4.0 (CC BY-NC). 10.1126/sciadv.1500988

Chad P. Byers,[1,2] Hui Zhang,[3] Dayne F. Swearer,[2] Mustafa Yorulmaz,[2] Benjamin S. Hoener,[2] Da Huang,[2] Anneli Hoggard,[2] Wei-Shun Chang,[2] Paul Mulvaney,[4] Emilie Ringe,[2,5] Naomi J. Halas,[1,2,3,5,6] Peter Nordlander,[1,3,5,6] Stephan Link,[1,2,6] Christy F. Landes[1,2,6]*

The optical properties of metallic nanoparticles are highly sensitive to interparticle distance, giving rise to dramatic but frequently irreversible color changes. By electrochemical modification of individual nanoparticles and nanoparticle pairs, we induced equally dramatic, yet reversible, changes in their optical properties. We achieved plasmon tuning by oxidation-reduction chemistry of Ag-AgCl shells on the surfaces of both individual and strongly coupled Au nanoparticle pairs, resulting in extreme but reversible changes in scattering line shape. We demonstrated reversible formation of the charge transfer plasmon mode by switching between capacitive and conductive electronic coupling mechanisms. Dynamic single-particle spectroelectrochemistry also gave an insight into the reaction kinetics and evolution of the charge transfer plasmon mode in an electrochemically tunable structure. Our study represents a highly useful approach to the precise tuning of the morphology of narrow interparticle gaps and will be of value for controlling and activating a range of properties such as extreme plasmon modulation, nanoscopic plasmon switching, and subnanometer tunable gap applications.

INTRODUCTION

Active control of the optical and electronic properties of nanoparticles is critical for many future technological applications, yet remains largely elusive. Optical tunability of metallic nanoparticles can be achieved by exploiting the sensitivity of the localized surface plasmon. Researchers in pursuit of in situ active control have used electronic (*1*, *2*), chemical (*3*), and electrochemical (*4*–*9*) approaches. Recent electrochemical efforts have resulted in either small reversible modulations (*7*, *8*) or large irreversible plasmon shifts (*10*, *11*). One strategy toward achieving

both isolated Au nanoparticles and strongly coupled Au nanoparticle dimers. Large changes in nanoparticle optical spectra are typically irreversible, but here we demonstrate a simple and robust chemical mechanism that allows large reversible changes in nanoparticle optical properties. We tune the optical and electronic properties by electrochemically controlling the nanostructure morphology, chemical composition, electronic coupling strength, and, ultimately, fundamental coupling mechanism. We initially demonstrate reversible tuning of the localized surface plasmon resonance of individual core-shell Au/Ag nanoparticles through

Downloaded from http://advances.sciencemag

materials
Ag
Au
Cu
Pt
anatase
cadmium
graphene
hematite
silver
copper
gold
zinc
tin
titanium
palladium
platinum
silicon
selenium
polyaniline
oxide
dioxide
selenide
nitride
arsenide
telluride
sulfide
phosphide
nanowire

Figure 1. Screenshot of the Fast Annotator showing the highlighting of gazetteer list and the general layout of the program.

are studied through an experiment with 43 unexperienced annotators.

METHODOLOGY

The objective is to create a tool for faster annotation (Fast Annotator) that will not compromise on normal accuracy (Manual Annotation). The document corpus that is used in this experiment is composed by documents retrieved from the web. Since these documents are only filtered by vocabulary, there are multiple types of documents present in the corpus. Because of the importance of the validity of content of the document, the relevant documents are only going to be composed of scientific peer-reviewed papers. Since these types of documents often require publication standards, which often includes a structured layout, we expect the relevant documents to be well-formatted. To classify these documents, verification of the layout is typically an easy task for a human annotator. To take advantage of this, first, the annotator has to determine whether the document has the aspect to be a scientific paper. Therefore, our classification categories can be separated in papers and non-papers. In the case the document is a scientific paper, the annotator still has to determine the relevance to the content, synthesis of nanoparticles. This process cannot be automated since the input source files are in PDF, and therefore many of the metadata found in these source files are oriented for printing or rendering purposes rather than reading and classifying.

In the case the document is not a scientific paper, it is automatically considered not relevant; however further refinement of the class label is needed. The purpose of this subsidiary classification task is to help identify low-level features and those that can be identified by modern feature extraction algorithms, such as deep learning autoencoders.

There are three sub-categories for helping determine the type of document: *poster/presentation*, *form*, and *other*; these are the subclass labels The poster/presentation category has documents that can be described as graphics, informational posters or presentation talks. The form category are all documents that are online application forms, survey or journal petitions. The third and final category, contains all documents that cannot be classified as any of the above, along with any documents that are not in the language of our research (English) since those are out of our scope. Documents that are scientific posters or presentations, but whose content is relevant, are considered not relevant for the purpose of simplifying the task for human annotators and validation of content.

Manual Annotation

To evaluate the performance of the Fast Annotator we have to compare it with the standard way of classifying documents without an annotation tool. We are calling Manual Annotation the classification of documents without the Fast Annotator. We are considering the time it takes the annotator to open, mentally determine the classification of the document, and physically classifying it. This Manual Annotation depends totally on the procedure in which the annotator classifies the documents. Because of this, we have created an algorithm that is to be followed by all the annotators.

Using an online stopwatch, the procedure to classify a document is to start the time, then open the document in the default PDF renderer for the machine. Once the document is classified, the annotator would move the file to the correspondent directory and pause the time. This ensures that the time for decision making and physical annotation is taken into account, while also following the way the Fast Annotator records its time.

Fast Annotator

The Fast Annotator (Figure 1) was designed with loose implementation of the classical Nielsen heuristics [9]. While designing the Fast Annotator, questions such as consistency of the user experience, feedback of user's input, simplicity, shortcuts and other heuristics where considered. For consistency purposes, all papers are shown to the user the same way: first page is in the central window, and the other pages (up to five) are shown as thumbnails on the left side of the screen. The thumbnails have two purposes, to help the user look ahead in the annotation process by showing a preview of the pages, and to aid the user to get familiar to the UI, since thumbnails is a very common aspect of many document readers, visually, the user can start with something similar to their previous experience and move to a new experience as they follow to the right. Only the first 5 pages of the document are shown. This is to increase the speed of PDF rendering with the hope of decreasing the final time for annotation.

The Fast Annotator shows the status of the annotation process (paper *i* out of *n*) on the bottom left side of the screen and every time a paper is classified, a "loading" message appears to let the user know that the operation is processing. These gives the user a sense of task progress as the user can see how many papers are done, and a sense of feedback speed as the loading message appears right after any classification button is pressed.

The Fast Annotator shows the gazetteer list as "Keywords list" and highlights all the words inside the document. The button layout is designed to show the difference in the types of documents (whether the document is a paper or not and then further classify based on relevance).

The procedure to use the Fast Annotator is simpler for the annotator than the Manual Annotation. Normally, the user annotating has to start the program and choose the directory were all the documents are located, but for the experiment this location was predetermined, so the user only had to start the program. Since the program keeps track of the time spent on each document in the background, the user does not have to keep track of the time as they had to in the Manual Annotation. Once the program starts, it queues all the documents in the specified directory, so the user does not have to open each file. The user simply has to click the category to classify the document, and the next file will be queued by the program right away.

Preliminary Experiment Design: Best-of-3, Large Batch

In a preliminary exploratory experiment to assess the feasibility of learning to filter from text features for the materials informatics domain, we created two large batches of files for testing Manual Annotation and an earlier version of the Fast Annotator. The earlier version of the Fast Annotator is functionally the same, with the difference that it has a few more button categories, and visually, the button layout is located on the left side of the screen rather than on the right on the current version. Each large batch contained 1260 files, consisting of 12 smaller batches of size 105 each (the least common multiple of 3, 5, and 7, for ease of experimenting with Best-of-3, Best-of-5, and Best-of-7 inter-annotator agreement).

Training data for supervised inductive learning was generated by creating a bag of words representation of 7633 unique tokens occurring in all small batches, after stop word removal and stemming.

In this and other preliminary experiments, we noted that the variance of annotation time for 1 to 3 annotators was high, suggesting that an experiment using 20-50 annotators would be more conducive to testing the hypothesis that the Fast Annotator required less user time than Manual Annotation, without loss of precision, recall, and accuracy.

Speedup Experiment Design: Best-of-43, Small Batch

For this experiment conducted using 50 documents and a participant pool of 43 users, the focus was on assessing speedup. **Ground truth was designated to be the previous annotation given by one of two subject matter experts.**

As described earlier, the fast annotator was designed to retrieve results at a more accelerated rate than doing the classification manually. The background information, layout, and survey were considered when organizing the design.

The subjects were volunteers Kansas State University industrial engineering students. They had no background knowledge about synthesis of nanomaterials, or how to use our annotation tool.

When preparing for the execution of the experiment, the information provided to our subjects was observed for accurate measurements when categorizing the data. A background summary of our project was provided on the creation of nanomaterials and how their annotations would be used in a normal environment as training data. Their objective was to complete the annotation as efficiently as possible. Following the definition and reasoning for the different categories described earlier: relevant, irrelevant, form, poster, and other.

Later, half of the students started with the Fast Annotator and the other half started with the Manual Annotation. This separation is to account for the learning curve of annotating a topic that the subjects were not experts in.

The task for each annotator was to annotate a total of 50 documents for each type of annotation. Each document corpus was previously annotated by experts in the field. The document corpora had equal representation of document categories for both the Manual and the Fast Annotator.

After the experiment, students were asked to take a completely confidential survey. This survey started with questions that analyzed the outcomes of the data, then later provided feedback on improvements to the Fast Annotator.

RESULTS

Preliminary Experiment: Best-of-3, Large Batch

In the preliminary experiment, the focus was on generalization quality rather than on the statistical significance of speedup in the annotator. Tables 1 and 2 show the results: accuracy, weighted average precision, average recall, F1 score, and area under the (receiver operating characteristic or ROC) curve, under 10-fold cross-validation, for Manual Annotation and the Fast Annotator. **Bold face** indicates the better of the two sets of results.

Table 1. Results for Manual Annotation.

Inducer	Acc	Prec	Rec	F1	AUC
Logistic	**75.2%**	0.711	**0.752**	0.709	0.640
J48	**78.3%**	0.782	**0.784**	**0.783**	**0.688**
IB1	79.9%	0.788	0.799	0.792	0.712
NB	**74.2%**	**0.790**	**0.742**	**0.757**	0.759
RF	79.4%	0.801	0.795	0.736	0.841

Table 2. Results for the Fast Annotator.

Inducer	Acc	Prec	Rec	F1	AUC
Logistic	69.3%	**0.764**	0.693	**0.719**	**0.664**
J48	77.9%	**0.785**	0.779	0.782	0.668
IB1	**83.8%**	**0.824**	**0.838**	**0.827**	0.695
NB	71.7%	0.789	0.718	0.742	**0.785**
RF	**83.3%**	**0.825**	**0.833**	**0.788**	**0.862**

The inducers compared in [1] were:

- Logistic: Logistic Regression
- IB1: Nearest Neighbor
- NB: Discrete Naïve Bayes
- RF: Random Forests

The average time required for Manual Annotation was 18,413.4 seconds versus 5,246.8 seconds for the Fast Annotator – a 251% speedup – with statistically insignificant gains in precision or AUC, slightly lower accuracy, and lower recall.

Speedup Experiment: Best-of-43, Small Batch

As described in the design specification, accuracy was assessed based on user annotations relative to expert ground truth. For N = 43, the accuracy of Manual Annotation classifications is 0.639 ± 0.125 (mean 0.639, stdev 0.125), while the accuracy of Fast Annotator classifications is 0.726 ± 0.114. The null hypothesis that the Fast Annotator is less accurate that Manual Annotation is rejected with $p < .00002071$ (2.071 × 10-5) at the 95% level of confidence using a paired, one-tailed t-test. Meanwhile, for N = 42 (due to one misrecorded time for participant #23) the time taken to process batches of 50 documents using Manual Annotation is 1070.41 ± 361.45 while the Fast Annotator time is 663.77 ± 468.14. The null hypothesis that the Fast Annotator is slower than Manual Annotation is rejected with $p < .0000537$ (5.37 × 10-5) at the 95% level of confidence using a paired, one-tailed t-test.

We received good feedback from the survey with 89.74% of the users indicating that highlighting the keywords helped them determine the type of document. We also found that on average, 97.44% only needed the first 3 pages to classify the document.

CONCLUSIONS

Summary of Results

The speedup trend observed in the preliminary experiment is upheld with lower variance but a much lower margin of victory: a 38% gain in speed using the Fast Annotator. Observed over 43 participants, however, the accuracy of the Fast Annotator is **also** conclusively higher.

The positioning of buttons, reduction of classification categories and overall layout along with the highlighting of keywords can account the increase in accuracy as the ease of use and learnability may have affected annotators' abilities to make a category classification decision. This may also be attributable to prior background expertise and interest.

Future Work

One priority in this continuing work is to isolate improvements to the user interface, such as highlighting and document previewing, from other causes of speedup and increased filtering precision and recall. These other causes include **UI-independent** optimizations such as document pre-fetching. It is important to be able to differentiate these causes to fairly attribute the observed improvement in performance measures for the system.

Attributing annotation speedup to specific user interface changes versus user-specific causes is a challenging open problem. To provide a cognitive baseline, collecting and analyzing survey data regarding annotators' expertise and interest in the domain topic could reveal an effect on the speed and accuracy of the results. A related problem is that of accounting for user expertise as subject matter experts and experience with the fast annotator: in our earliest experiments [1], we obtained greater speedups (251% as mentioned above) that may be attributable to greater familiarity with the fast UI due to the original annotators being UI developers. Although the hypothesized trends were supported by the experiment reported in this paper, using novice participants who were given only a rubric, these trends are lower in magnitude and significance. We hypothesize a learning curve that may be useful to model.

Further analysis for user interface design is planned for the Fast Annotator. To draw conclusions and test new tools, a technology such as gaze tracking and gaze prediction [5] could be used to expand the features available for relevance determination, and also to personalize and tune the interface for faster response. One particular application of this technology is to procedurally automate layout of annotation interface elements for user experience (UX) objectives, particularly the efficiency of explicit relevance feedback and multi-stage document categorization.

Information extraction from text and learning to filter documents (especially from text corpora) are already actively-studied problems in different scientific fields, and our project aims to aid in this area. As technology progresses, however, machine learning for information retrieval, information extraction, and search is being applied to more types of media, such as video and audio. An efficient video or audio annotator would increase the range of application of enabling technologies, such as action recognition, to different fields.

Finally, we are investigating applications of this type of human-in-the-loop information filtering in other problem domains, such as network traffic monitoring in cyberdefense, and anomaly detection. We hypothesize that reinforcement learning to develop policies for UI personalization can yield improvements in filtering quality such as the kind reported in this paper.

ACKNOWLEDGMENTS

The authors thank Yong Han and David Buttler of Lawrence Livermore National Labs for helpful feedback and assistance with ground truth annotation, and Tessa Maze for help with editing.

This work was funded by the Laboratory Directed Research and Development (LDRD) program at Lawrence Livermore National Laboratory (16-ERD-019). Lawrence Livermore National Laboratory is operated by Lawrence Livermore National Security, LLC, for the U.S. Department of Energy, National Nuclear Security Administration under Contract DE-AC52-07NA27344. This work was also originally supported in part by the U.S. National Science Foundation (NSF) under grants CNS-MRI-1429316 and EHR-DUE-WIDER-1347821.

REFERENCES

1. Aguirre, C. A., Gullapalli, S., De La Torre, M. F., Lam, A., Weese, J. L., & Hsu, W. H. (2017). Learning to Filter Documents for Information Extraction using Rapid Annotation. *Proceedings of the 1st International Conference on Machine Learning and Data Science.* IEEE Press.
2. Baumgartner Jr., W. A., Cohen, K. B., Fox, L. M., Acquaah-Mensah, G., & Hunter, L. (2007). Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics, 23*(13), i47-i48. doi:10.1093/bioinformatics/btm229
3. Ciravegna, F., Dingli, A., Petrelli, D., & Wilks, Y. (2002). User-System Cooperation in Document Annotation Based on Information Extraction. *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2002): Lecture Notes in Computer Science 2473* (pp. 122--137). Berlin, Germany: Springer.
4. Fiorini, N., Ranwez, S., Montmain, J., & Ranwez, V. (2015). USI: a fast and accurate approach for conceptual document annotation. *BMC Bioinformatics, 16*(83). doi:10.1093/bioinformatics/btm229
5. Karaman, Ç. Ç., & Sezgin, T. M. (2017). Gaze-based predictive user interfaces: Visualizing user intentions in the presence of uncertainty. *International Journal of Human-Computer Studies, 111*, 78-91. doi:10.1016/j.ijhcs.2017.11.005
6. Kim, E., Huang, K., Saunders, A., McCallum, A., Ceder, G., & Olivetti, E. (2017). Materials Synthesis Insights from Scientific Literature via Text Extraction and Machine Learning. *Chemistry of Materials, 29*(21), 9436–9444. doi:10.1021/acs.chemmater.7b03500
7. Lang, K. (1995). NewsWeeder: Learning to Filter Netnews. *Proceedings of the 12th International Conference on Machine Learning (ICML 1995)* (pp. 331-339). San Francisco, CA, USA: Morgan Kaufmann.
8. Müller, C., & Strube, M. (2001). MMAX: a tool for the annotation of multi-modal corpora. *Proceedings of the 2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, (pp. 45-50).
9. Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 1994)* (pp. 152-158). New York, NY, USA: ACM. doi:10.1145/191666.191729
10. Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002). Methods and Metrics for Cold-Start Recommendations. *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development* (pp. 253-260). New York, NY, USA: ACM.
11. Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., & Tsujii, J. (2012). BRAT: a Web-based Tool for NLP-Assisted Text Annotation. *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (ACL 2012): Demonstrations* (pp. 102-107). Association for Computational Linguistics.