Polymer Discovery

Thomas Goddard, Konstantin Litovskiy, Nathan Nichols-Roy,
Matthew Reed, Igor Shvartser, Nicholas Smith, David Zeppa

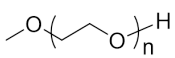
Baskin School of Engineering, UC Santa Cruz & IBM Almaden Research Center, San Jose, CA

ABSTRACT

The recognition of polymer images in the literature and patents is key for automatic understanding of the wealth of polymer data already known. While tools such as OSRA (Optical Structure Recognition Application) [1] exist to identify and interpret chemical structure images, these tools do not yet work for polymers. Our tool, Polymer OSRA(P-OSRA) extends OSRA's capabilities by being able to recognize brackets and parentheses in chemical diagrams. To date P-OSRA pre-scans the image; records and alters the image by removing brackets and parentheses from the diagram; calls OSRA and then collects and edits the resulting SMILES[2] string from OpenBabel[3] to reflect changes needed for describing a polymer image. P-OSRA then populates a datamodel to allow for subsequent querying of polymer substructures (such as repeat units).

INTRODUCTION

Accelerated materials discovery is at the core of innovation, economic opportunities, and global competitiveness. The research process is responsible for bringing new materials to market. In 2011, President Obama launched the U.S. Material Genome Initiative (MGI) and challenged researchers, policy makers, and business leaders to reduce the time and resources needed to bring new materials to market—a process that today can take 20 years or more. There is great potential in leveraging modern data mining, big data analytics techniques, and physics based modeling (high performance computing) to significantly shorten the Research & Development cycle in material sciences. Polymers, as an important part of materials science, are the focus of much research for applications in the areas of semiconductors (e.g. low-k dielectrics, photolithography, and directed self-assembly), nanomaterials, polymeric drug delivery vehicles, desalination membranes, and recyclable polymers for green chemistry. In order to accelerate polymer discovery, the first challenge is to automate the retrieval of all polymers made to date both in the literature and in patents. Although there are standardized nomenclatures for small molecules, standardized nomenclature for polymers is not straight forward and it is often more informative to represent these molecules as molecular structure diagrams or acronyms (e.g. PEG). Thus, this project's goal is to identify polymer images in published research, analyze them and output the structure to a datamodel that allows their substructures to be queried.



BACKGROUND
OSRA (Optical Structure Recognition Application)

P-OSRA extends OSRA, a tool for parsing images of chemical structures in documents, to support *polymer diagrams* that use bracket and parentheses notation in their representations. P-OSRA also offers a data interface to allow storage and querying of repeat units and end groups (sub-components of the polymers). This is important because

> "Proliferation of computer technologies has brought forward the necessity of new data formats to exchange information in a machine-readable way within the context of a scientific publication … Our approach to recovery of chemical information from published material is to reuse to the fullest extent possible the existing software created by the open source community and to invite further development and participation by releasing our work as free and open source … OSRA has been designed with a wide range of applicability in mind: it does not rely on the document image being of any particular resolution, color depth, or having any particular font used. To manipulate images, OSRA employs the ImageMagick library [4] that allows parsing of over 90 different image formats, including the popular TIFF, JPEG, GIF, PNG, as well as Postscript and PDF…"[5, page 1].

ACKNOWLEDGEMENT

Many thanks and appreciation for IBM Researchers Dr. Julia Rice, Dr. Hans Horn, and Dr. Amanda Engler; creator of OSRA, Dr. Igor Filippov; & IBM Researcher Scott Spangler.

RESULTS

High Level Process Diagram of P-OSRA

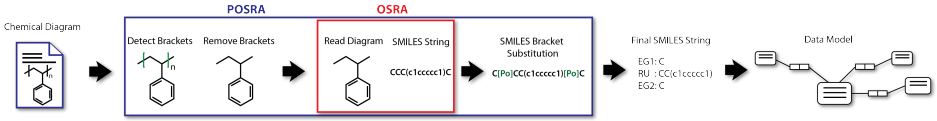


Figure 1

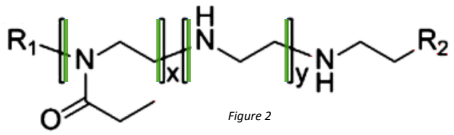


Figure 2

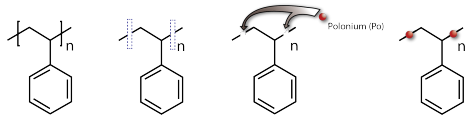


Figure 3

Finding and Removing Brackets

At first, we took the approach of using point density data from a vector conversion of a structure image to pinpoint segments of the diagram that could be brackets. This proved mostly inaccurate and difficult to implement. Next, we tried an algorithm to detect endpoints in the diagram and found this to be promising. This algorithm works by checking each pixel's surrounding adjacent pixels to see if it is an endpoint. We then look for pairs of endpoints that properly align, check for reasonable spacing and bond intersection, and tentatively store the coordinates of these endpoints as the location of a single bracket. From these coordinates, we draw a bounding box around the bracket and color all pixels inside the box white, removing the bracket, and then redraw the bond without the bracket. Figure 2 is the resultant image after tracing, in green, endpoints that are believed to be brackets. Figure 3 depicts the polymer polystyrene and the process in which we alter the image. After detecting the endpoints of the image and finding the brackets, we are able to box out the brackets entirely and replace them with placeholder atoms (we chose the bi-valent Po atom as it is highly unlikely to appear in any polymer structure).

Editing the SMILES String

Since we have replaced the brackets by placeholder atoms, OSRA then continues to parse the image as a normal chemical structure (red box in Figure 1). The resulting SMILES string (obtained by calling OpenBabel) reflects this structure. But since the image represents a polymer, the generated SMILES string needs to be edited by splicing it (at the locations of the Po placeholder atoms) and formatting it into end groups and repeat units. Figure 4 is a chemical diagram of a polymer poly(trimethylene carbonate) and it has been color coded to distinguish end groups ("EndGroups 1,2") from repeating segments ("RepeatUnit"). Figure 5 is the resulting SMILES string after OSRA/OpenBabel has completed parsing the diagram (top) and the final edited SMILES strings for the polymer (bottom).

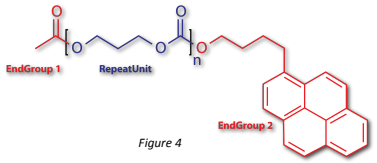


Figure 4

CC(=O)[Po]OCCCOC(=O)[Po]OCCCCc1ccc2ccc3cccc4ccc1c2c34

EG1: CC(=O)X1
RU: X1OCCCOC(=O)X2
EG2: X2OCCCCc1ccc2ccc3cccc4ccc1c2c34

Figure 5

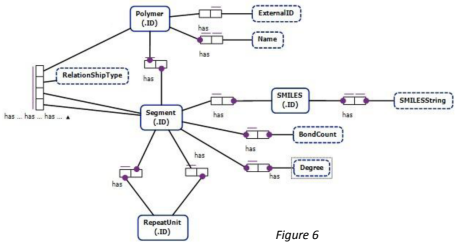


Figure 6

Creating a Data Model

We have designed a data model in Object Relational Model format (ORM)[6] so that we can store and query the images. With a queryable database, researchers can search for particular substructures or "similar" repeat units that they may desire to include in a new polymer. Retrieval of these units will allow them to rapidly assemble and test new polymers via computer simulation and eventually in the lab. ORM is very flexible and can output to many database formats including MySQL[7]. Figure 6 shows an ORM diagram tree, graphically showing the relationship between substructures and information about the polymer. For example, a Segment may have exactly one SMILES identifier, which has exactly one SMILES string, or in a more complicated example a Segment might have one or more RepeatUnits, that may or may not be nested within each other.

INITIAL CHALLENGES

Understanding the Code

A publication by OSRA's creator, Igor Filippov, helped us to understand the basic function of the program giving us insight into which libraries we needed to be familiar with, but it did not reveal the underlying structure or details of the code. Because the OSRA program code uses many image processing and chemistry terms that we were not familiar with, we spent many hours in code reviews, compiling internal documentation, and generating a function tree using Doxygen[8]. Using these software engineering tools and techniques, we were able to identify locations for code insertion while preserving the functionality of OSRA.

Understanding the Fundamentals of Chemistry

Initially the team was given basic chemistry data sheets, which touched on electron density and orbital theory, to determine the chemistry knowledge the team had at the start of the project. From the questions and concerns of the team, it was clear that there were many inadequacies in the team's overall knowledge of organic chemistry including basic terminology. Thus, a lecture on nomenclature of organic families was given by Drs. Rice, Horn, and Engler to build the necessary organic chemistry knowledge base the team would need to use to communicate during the project (personal communication, January 2014).

Creating a Test Harness

One of our goals in building the P-OSRA extension was to show that individual features of the program worked correctly. Unit testing allows us to isolate each part of the program and find problems early in the development cycle. First, we collected data including a set of selected images and their corresponding SMILES strings that our university team and IBM advisors believed to be appropriate for testing. We then wrote test scripts to automate processing on the selected images. This process not only makes development more efficient, but it also helps us benchmark and optimize the code to preserve OSRA's power.

PERFORMANCE

OSRA has been parallelized with OpenMP[9], allowing the program to be multithreaded. Since the program involves mostly image processing requiring multiple passes over images at different resolutions, OSRA lends itself to parallelization allowing batch processing to be more efficient. With our additions, we hope to preserve the same level of performance and plan to better incorporate OpenMP in our image pre-processing stage to maximize efficiency.

CONCLUSIONS

We have created an extension to OSRA, making it more comprehensive, that provides the ability to parse polymer chemical diagrams. Our plans include the addition of more comprehensive and detailed documentation and development tools to OSRA to make it easier for other teams to extend OSRA's capabilities. POSRA is written in C++ and is currently supported on Windows and Linux systems, both 32 bit and 64 bit. We plan to release POSRA as open source software under the public domain.

REFERENCES

1. I. V. Filippov, "OSRA: Optical Structure Recognition Application", <http://cactus.nci.nih.gov/osra/>, N.p., n.d. Sept. 12, 2009.
2. D. Weininger. "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules". *J. Chem. Inf. Model*, 1988, 28, 31
3. Noel M O'Boyle, Michael Banck, Craig A James, Chris Morley, Tim Vandermeersch and Geoffrey R Hutchison. "Open Babel: An open chemical toolbox".J Cheminf, 2011, 3, 33
4. "Convert, Edit, and Compose Images." *ImageMagick: Convert, Edit, Or Compose Bitmap Images.* N.p., n.d. Web. 02 Apr. 2014.
5. I. V. Filippov, M. C. Nicklaus, "Optical Structure Recognition Software To Recover Chemical Information: OSRA, An Open Source Solution" J. Chem. Inf. Model., 2009, 49, 740
6. T. Halpin, "Object Role Modeling." <www.orm.net> N.p., n.d. Feb. 28 2014.
7. "MySQL :: The World's Most Popular Open Source Database." MySQL :: The World's Most Popular Open Source Database. N.p., n.d. Web. 01 Apr. 2014.
8. D. van Heesch. "Doxygen." <www.doxygen.org>. N.p., n.d. Feb. 24. 2014
9. "Get." OpenMPorg RSS. N.p., n.d. Web. 01 Apr. 2014.