Welcome to the second decade of online taxonomy construction and maintenance!

In the early days of online taxonomies, our focus was primarily on navigation. We were using online taxonomies to improve the browsing experience on websites, and occasionally, when good metadata was available, we used taxonomies to improve search. But we’ve come a long way since then, and our focus has changed. We now build taxonomies to support auto-classification systems.

The early efforts are exemplified by the categories you see on e-commerce websites like eBay or Amazon; these sites include high-level, general-to-specific categories that allow users to drill down, or navigate, to exactly the product they need.

When we started creating online taxonomies, auto-classification was in its infancy, and the software was not really reliable yet. There was a huge disconnect between the high-level structure of a navigational taxonomy and the data requirements of auto-classification software.

More often than not, early auto-classification software seemed to be overly ambitious. Many early attempts involved clustering techniques that were used to collect documents that included similar “bags of words.” The most unique words often occurred at the very specific detail level, so the clusters tended not to make sense on their own. They required a suitably descriptive, higher-level label.

But technology has evolved, and we are in a different situation today. With the advent of sophisticated content analytics techniques, entity extraction is increasingly reliable, and we now routinely use proper noun entities, as well as noun phrases, to enhance our classification schemes.

Taxonomy building has also changed as a result of these advances in entity
and phrase extraction techniques. When we build taxonomies now, we still work to create descriptive high-level categories, but we also work to accurately associate the lower-level entities with the higher-level categories we’ve created. We provide an overview of this taxonomy building process here, although each step in the process really deserves an article in its own right.

Entities and Text Mining

Entities are phrases that identify people, places and things that are easily recognized by content analytics software. Entities are often recognized by specific patterns in text. Place names are good examples of entities, but entities can also be based on lists of terms that already exist, such as programming languages (Figure 1).

Entities often supply the best evidence for identifying higher-level taxonomy tags, and text-mining tools are now very adept at identifying entities in content. You’ll likely use text-mining tools at varying points in your taxonomy development cycle.

For example, you might want to use a text miner to identify product names in your content. In this case, your highest-level taxonomy category would be Product, and you would likely also have a metadata tag, possibly called “MyCompanyProducts,” that is associated with your content. The values for the “MyCompanyProducts” tag are drawn from the terms in the Product taxonomy. You might also include second level categories that describe product types.

If your taxonomy supports a high-tech site, your secondary tags could be Software Products, or Open Source Software, followed by specific product names, like Lucene. You probably wouldn’t include the high-level terms as actual values in your metadata tags, since they are too general, but you would include them in your taxonomy so that you can identify what kind of thing the entity is.

Most text-mining tools provide information about how many of each type of entity they find. If, for example, you run a subset of your content through a text-mining tool and see many specific Product names identified, you can assume that Product will be a good high-level descriptive category.

Text miners can often identify phrases, and these phrases are useful during the early taxonomy design. If you see many phrases that describe specific activities around a subject area, for example, programming, software engineering,
coding or software testing in your content, then “Software Development” is a good candidate value for a Subject or Topic tag. You’ll also use these phrases later, as synonyms, so the auto-classification tools pick up term variants in your content.

A Basic Methodology

No matter which tool you use, you should follow this basic methodology when building your taxonomy.

1. Understand the auto-classification technique that your tool uses.
2. Create a taxonomy model.
3. Test your classification.

Your final step describes the way your classification will be used, and this step comes down to cases. If you use your classified content to drive a faceted search user interface (UI), for example, you’d add an additional step to test your classification in the context of the new UI.

Understanding Auto-categorization Techniques

Understanding how the various auto-classification products work is key to building an effective taxonomy for categorization. Most focus on one technique, with secondary support for the others, and the best products provide a hybrid approach.

Auto-classification products rely on a good classification model. The model includes the high-level categories, or tags, that will be associated with content and also defines the evidence, or rules, that determine when these tags will be applied. The tools for collecting and analyzing the evidence differentiate the products.

You can often judge the strength of any vendor’s approach by looking at the richness of its taxonomy management offering. Look at the tools that are provided to help you create your classification and also at the tools that are provided to help you test your taxonomy. These tools are essential, but they vary widely by vendor, and robust test tools often indicate a more robust product.

The tools vary in terms of the amount of maintenance required to create an accurate classification. All of the techniques require a good amount of up-front design work and ongoing testing and maintenance. No auto-

classification tool is perfect, but software classification is always more consistent than human categorizers.

Linguistic/Lexical Tools. Tools that concentrate primarily on a lexical approach will have the richest taxonomy management functionality. Lexical tools let you gather and rank representative words and phrases that are associated with the concepts to be classified; these tools allow you to identify the keywords and phrases the way they occur within the text.

You rely heavily on identifying synonyms and term variants when you use the lexical tools, since you want to identify the various ways that important concepts appear in your content. Often you don’t want to identify only exact phrases, but also words that appear in the same sentence. Products often provide settings that help.

Lexical tools require a great deal of initial up-front analysis and a very iterative development and testing cycle. You will add terms, synonyms and new rules as you notice different occurrences of the important phrases in your content.

Rules-based Tools. Rules-based tools provide a rich syntax that you use to control the way the evidence in the taxonomy will be used to add tags to text. As with lexical tools, explicit rules can be used to indicate that you want words to appear in the same sentence, rather than only as a phrase.
But rules are often used to identify words and phrases that should not be used as evidence. Rules-based systems also allow you to compare more than one lexical construct to provide an additional level of control over the classification. They can be used to tell the difference between, or disambiguate, similar terms or entities that appear in your taxonomy.

For example, there is a Will Smith who plays American football for the New Orleans Saints, and there is also Will Smith, the actor. If the entity “Will Smith” appears in a sentence, the evidence would indicate that the content could be categorized as either Entertainment or Sports. But if the title *Men in Black* also appears in the same sentence, the sentence is more likely talking about Will Smith the actor and should be classified as Entertainment. Rules allow you to compare the entities and make the appropriate tagging determination.

Right now, there is no common syntax for developing rules; the syntax varies from tool to tool. Rules syntax ranges from the kind of Boolean syntax that is usually associated with search engines to the more complex syntax more commonly used in programming languages. Because of this lack of consistency, the people who create and maintain these rules will have a more specialized skill set and will require more training.

**Machine Learning and Predictive Analysis Techniques.** Machine learning and predictive analysis techniques are interesting because they often initially appear to be less manual-labor-intensive than the other approaches. Although many of these systems claim that they are not using taxonomies on the back end, they are still identifying and classifying phrases and entities.

These tools take advantage of the same content analysis and text-mining techniques that taxonomists use to enrich classification schemes, but their models are more complex than a traditional taxonomy structure. And these systems rely on iteration to continuously validate the models they use. Again, you may not create a traditional hierarchical taxonomy to support machine-learning systems, but you will create taxonomies that these systems can use for reference. You might also select representative document sets to train these systems.

Maintenance of machine learning systems always involves repeated training, especially when you add new content. You will also help revise the larger machine-learning model as you learn more about your content.

**Create a Taxonomy Model**

Designing a model for auto-classification is similar to but not exactly the same as designing a taxonomy for manual indexing or navigation. Like more traditional taxonomy construction, both top-down and bottom-up analysis is required, and knowledge of the content to be classified is key.

Typically, you figure out how you want to organize your taxonomy, and then you build it out. You do this task by deciding on the general categories first and then adding more specific terms within each category later. Your model can be hierarchical, and, as we said above, you typically associate the highest-level terms with the actual metadata tag names.

You’ll want to figure out the level of specificity of the terms to include in your tags. Do you want to tag major mentions only, or do you need to include every occurrence of a Product name? Again, there might be many terms in your taxonomy that are used as evidence and not as actual values that appear in your metadata tags.

**FIGURE 3. A Product hierarchy, including the general Software Development and Open Source Software terms and the very specific Lucene product name.**

You’ll often import terms from existing sources or use an existing taxonomy as a basis for your model for auto-classification. But while taxonomies that are used for manual indexing or for navigation can usually accommodate poly-hierarchy, your auto-classification taxonomy model should avoid it. Using the same term as evidence for two separate categories will lower the accuracy in each.
**Test Your Classification**

Testing your classification is the most important step in the process. Language is imprecise, no matter what subject area your taxonomy covers, and you will be surprised (and often amused) by some of the mistakes an automatic tool can make. You will want to use the test tool to look for inaccurate tags, but you will also want to look for tag omissions.

**Collect a Document Set.** It’s useful to collect a set of documents that you use over and over as you test. If you use the same document set for each test, you can easily track the impact of any changes you make. Since you eventually add content to this set, you can re-test and see how your classification performs as your content changes.

**Configure Your Auto-classification Tool.** Your configuration will depend on the tool you use, and it will usually require some support from your IT department, but you will want to have a basic knowledge of the default rules it will use to do its job. If your tool is basing its classification decisions on words that occur within sentence boundaries, for example, you’ll want to be sure that your content includes sentence boundaries that the tool will understand.

Iteration is key to the testing process, since minor changes in your taxonomy often have major impacts on your classification. No taxonomy is ever really complete, but you will want to understand when your classification is ready to be put into production.

So again, welcome to the brave new world of taxonomy development for auto-classification. While this new methodology has a bit more in common with software development than it does with traditional library science, the outcome is much more satisfying, and you will put your information management skills to good, practical use.