Supporting Subject Librarians with AI Solutions

Osma Suominen

IFLA Subject Analysis and Access WG on Automated Indexing Webinar

9 November 2022

16 40
NATIONAL LIBRARY
OF FINLAND

About me

Twitter:
@OsmaSuominen

LinkedIn:
osmasuominen

GitHub:
@osma

Osma Suominen
Information Systems Specialist, National Library of Finland

Doctoral thesis "*Methods for Building Semantic Portals*"
Semantic Computing Research Group, Aalto University, 2013
Supervisor Professor Eero Hyvönen

Joined the National Library in 2013
to set up the Finto.fi thesaurus and ontology service

Team leader for automated cataloguing project (since 2019)
Working on automated subject indexing (Annif, Finto AI)

Open source software projects e.g.:

- Skosify - Validation and QA tool for SKOS vocabularies
- Skosmos - SKOS vocabulary publishing tool
- Annif - Tool for automated subject indexing and classification

annif

developed since 2017

General purpose open source **tool** for automated subject indexing and classification

Multilingual, supports many vocabularies

Code on GitHub, website with test form and API

Global development and user community; user forum **annif-users** on Google Groups

annif.org

fintoai

launched in 2020

Automated subject indexing **service** for production use, based on Annif

Supports indexing with the General Finnish Ontology YSO in Finnish, Swedish and English language

Web user interface and API service

Intended to support subject cataloguers in Finland regardless of institution (GLAMs, public administration); sister project to the Finto vocabulary service

ai.finto.fi

Outline

1. Preparing the ground for AI solutions
2. Algorithms and data sets
3. Interfacing between developers and librarians
4. Putting AI into production

1. Preparing the ground for AI solutions

Setting expectations, communicating goals

What are you aiming for?

- improvement of subject cataloguing processes?
- indexing of large amounts of documents that humans can't handle?
- replacement of subject cataloguers by machines?

These are all different goals that you need to communicate to everyone involved

Humans vs. algorithms in subject cataloguing

Humans

Have background knowledge about the world
Remember what they've done in similar situations
Memorize core parts of the vocabulary
Are creative
Understand bias and try to avoid it

Are slow
Are inconsistent
Make (human) mistakes

Algorithms

May be trained on millions of examples
See patterns in data that humans miss
Know all of the vocabulary, but in a shallow way
Are fast and tireless

Are easily biased
Struggle with change
Don't really understand what they're doing
Make mistakes that don't make any sense

Machine-assisted vs. fully automated subject indexing

Machine assisted (semi-automatic)

Beginner friendly (e.g. student indexing thesis)
More consistent indexing
Possibly faster than without assistance

Users like it, but is it actually better?
Can we measure it?

Fully automated

Collections that can't be indexed manually
Crucial to set expectations accordingly
Quality not be as good as *professional* indexing - but maybe better than non-expert?

Buy or Build?

Commercial solution

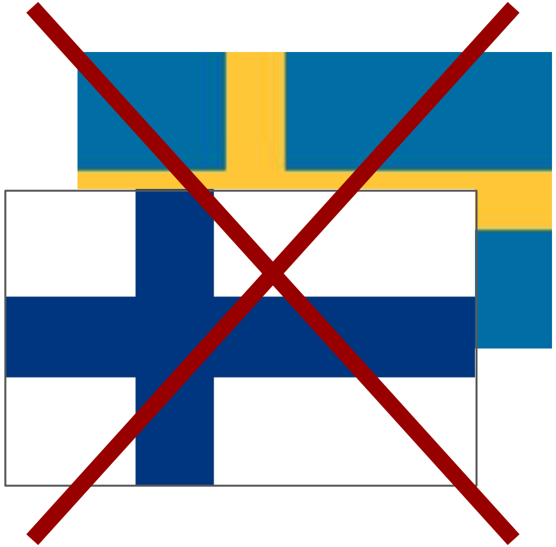
Pros: Apply existing, mature products
Access to expertise not available in-house
Clear responsibilities: provider & customer

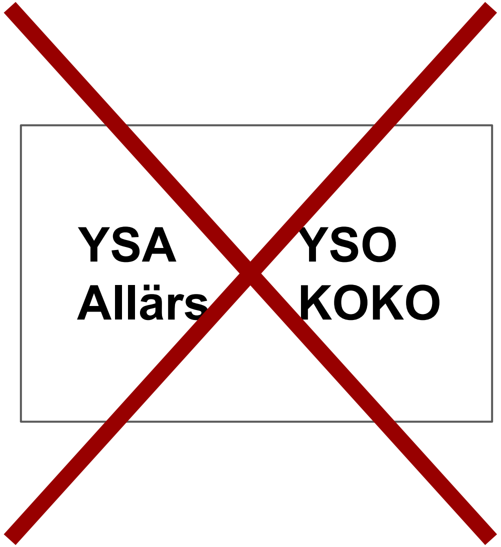
Cons: One size fits all solutions
Lack of options (e.g. language, vocabulary)
Vendor lock-in

DIY open source solution

Pros: Build solutions based on actual needs
Competency building for own staff
Community building & sharing

Cons: Requires dedicated staff
Requires in-house expertise
Sustainability?





YSA
Allärs

YSO
KOKO

black box

Required resources

For a successful automated subject indexing project, you will need:

1. a well defined subject vocabulary or classification
2. enough good quality training and evaluation data
3. staff with necessary skills [next slide]
4. computing resources (from laptops sometimes up to big servers)

Required staff skills

Collectively, your team should:

- know the subject vocabulary and how it's used
- be familiar with subject cataloguing practices processes
- be able to work with data sets, e.g. database dumps of text corpora
- be familiar with the tools for automated indexing
- understand evaluation metrics & methodology
- be able to operate production web services
- **talk to each other & people affected by automation**

Annif tutorial

Hands-on guide - arranged 5 times in 2021

SWIB22
Semantic Web in Libraries

ELAG RIGA '2022
LES FUTURS FANTASTIQUES

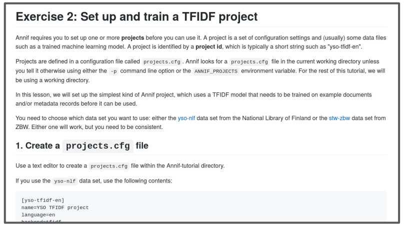
SWIB21
Semantic Web in Libraries

SWIB20
Semantic Web in Libraries

DCMI Virtual, 2020
September 14th-25th, 2020

SWIB19
Semantic Web in Libraries

annif tutorial
Data se
tutorial
16 40 THE NATIONAL LIBRARY OF FINLAND
ZBW Leibniz-Informationszentrum Wirtschaft Leibniz Information Centre for Economics



Exercise 2: Set up and train a TFIDF project

Annif requires you to set up one or more **projects** before you can use it. A project is a set of configuration settings and (usually) some data files such as a trained machine learning model. A project is identified by a **project id**, which is typically a short string such as "yso-tfidf-en".

Projects are defined in a configuration file called `projects.cfg`. Annif looks for a `projects.cfg` file in the current working directory unless you tell it otherwise using either the `-p` command line option or the `ANNIF_PROJECTS` environment variable. For the rest of this tutorial, we will be using a working directory.

In this lesson, we will set up the simplest kind of Annif project, which uses a TFIDF model that needs to be trained on example documents and/or metadata records before it can be used.

You need to choose which data set you want to use: either the yso-nlf data set from the National Library of Finland or the stw-zbw data set from ZBW. Either one will work, but you need to be consistent.

1. Create a `projects.cfg` file

Use a text editor to create a `projects.cfg` file within the Annif-tutorial directory.

If you use the `yso-nlf` data set, use the following contents:

```
[yso-tfidf-en]
name=YSO TFIDF project
language=en
backend=tfidf
```

Videos and exercises freely available on YouTube & GitHub!

16 40
NATIONAL LIBRARY OF FINLAND

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

2. Algorithms and data sets

Classification vs. subject indexing

Classification

Goal: Pick the **one correct class** among many defined classes that best fits this document

E.g. DDC, UDC, fields of science classifications

In machine learning: **multiclass** classification

Subject indexing

Goal: Pick **a few (3-12) concepts** from a subject vocabulary (subject headings or thesaurus) that best describe the topic of this document

E.g. LCSH, MeSH, AGROVOC

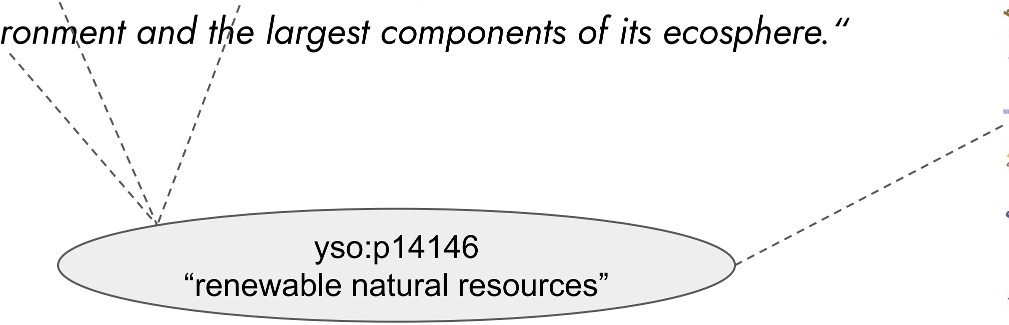
In machine learning: **multilabel** classification; with big vocabularies and messy, real world data sets → **extreme multiclass classification (XMC)**

Lexical vs. associative algorithms for subject indexing

lexical approaches (e.g.: MLLM, stwfsa)

match the **terms** in a document
to **terms** in a controlled vocabulary

*“**Renewable resources** are a part of Earth's **natural** environment and the largest components of its ecosphere.“*


yso:p14146
“renewable natural resources”

Lexical approaches need comparatively little training data.
Best suitable for multilabel subject indexing.

associative approaches (e.g.: SVC, fastText, Omikuji)

learn which **concepts** are correlated with which **terms**
in documents, based on training data

renew
energi
resourc
environment
sustain
develop
technolog
product
natur
sourc
photovolta
implement
system
manag
report
northern
solar
assess
pollut
practic
perspect
finland
model
chemic
research
challeng
approach
programm
carbon
forest
econom
potenti
sector
electr
power
local
polici
analysi
biomass
design
global
region
summari
opportun
architectur
commun
chang
hydrogen
agricultur
growth
fuel
thermal
project
impact
biotechnolog
book
wood
futur
rural
plan
action
action
gener
altern
russia
africa
oil
cell
new
india
water
plant
environ
applic
activ
land
measur
fibr
biofuel
econom
eu
ecolog
nativ
option
union
beyond
stock
case
affair
statist
handbook
execut
materi
capit
control
petroleum
compani
proceed
toward
end
solut
overlap
base
alberta

Associative approaches need a lot more training data in order to cover each subject.
Both for multiclass and multilabel classification.

Algorithms may be used **alone**, or in combinations, **ensembles**

Ensembles are nearly always better than individual algorithms

Make sure to have enough training and evaluation data

Collect already indexed documents, or metadata about documents, from

- bibliographic catalogues
- discovery systems
- institutional repositories
- digital archives

Ideally you should have

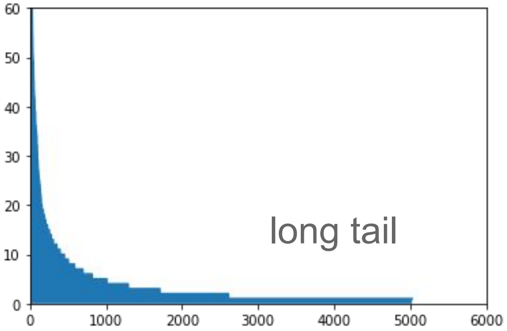
for lexical algorithms: 1000 or more indexed documents (or abstracts)
for associative algorithms: (10 * size of vocabulary) documents (or records)

Text: title, abstract, keywords, fulltext...

Text is the **main, often only**, information fed into automated subject classification algorithms. It is important to have enough good quality text that represents the topic.

Title	Often too short to capture the whole topic; can be figurative
Title + keywords	Better than title alone, even if keywords are uncontrolled
Abstract	Very good summary of the document
Fulltext	Good but may be noisy. Extracting text from PDFs or OCR processes can produce garbage. Often enough to use just the beginning (e.g. first 5000 characters)

Biases, omissions, quality errors



Many quality issues to watch out for:

- too few documents in a collection; skewed towards some topic areas
- existing subject indexing is inconsistent or has many errors
- few or no documents about emerging topics
- only 0-2 documents about many concepts in the vocabulary (long tail)

Some algorithms are more sensitive to these problems than others.
Extreme classification algorithms (e.g. Omikuji) are better than others.

3. Interfacing between developers and librarians

Workshops

We’ve arranged workshops at the biennial Library Network Days (2017, 2019, 2021) where participants performed subject indexing and/or rated suggested subjects for example documents. The subjects were produced either by human indexers or Annif algorithms.

The workshops have been very successful in spreading awareness about automated subject indexing among Finnish librarians.

2019 workshop. Photo: Mikko Lappalainen.

User testing of AI tools & services

Can be approached from many angles:

1. usability testing of user-facing tools (e.g. screen recording, think aloud protocol)
2. subject librarians make notes during their daily work
3. asking for user feedback via survey forms

We've done a little bit of 1., some more of 2. and 3.

Agile practices: librarians as users

Software & systems development is nowadays often done using agile methods.

Subject librarians can be active users in the process, for example:

- testing prototypes and intermediate versions
- suggesting and prioritizing features
- evaluating results of algorithms

Evaluation approaches (Golub et al. 2016), **emphasis** mine

1. Evaluating indexing quality directly through **assessment by an evaluator** or by **comparison with a gold standard**.
2. Evaluating indexing quality directly **in the context of an indexing workflow**.
3. Evaluating indexing quality indirectly through retrieval performance.

The different evaluation approaches are complementary.
Not a good idea to look at just a single measure.

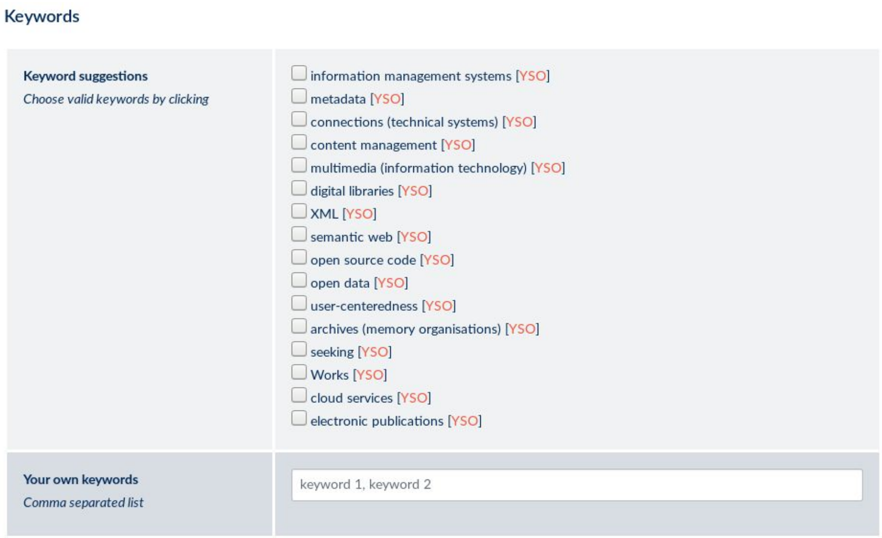
Golub, K., Soergel, D., Buchanan, G., Tudhope, D., Hiom, D., and Lykke, M. 2016. A framework for evaluating automatic indexing or classification in the context of retrieval. Journal of the Association for Information Science and Technology, 67(1): 3-16.

4. Putting AI into production

Deep vs. lightweight integration

Deep integration: automated topic suggestions in the cataloguing user interface

Lightweight integration: separate web UI, copy & paste strings in the correct format



Keywords

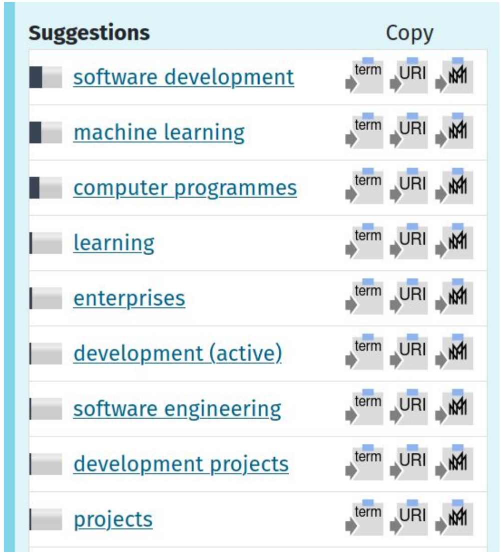
Keyword suggestions
Choose valid keywords by clicking

- □ information management systems [YSO]
- □ metadata [YSO]
- □ connections (technical systems) [YSO]
- □ content management [YSO]
- □ multimedia (information technology) [YSO]
- □ digital libraries [YSO]
- □ XML [YSO]
- □ semantic web [YSO]
- □ open source code [YSO]
- □ open data [YSO]
- □ user-centeredness [YSO]
- □ archives (memory organisations) [YSO]
- □ seeking [YSO]
- □ Works [YSO]
- □ cloud services [YSO]
- □ electronic publications [YSO]

Your own keywords
Comma separated list

keyword 1, keyword 2

JYX Dspace repository using Finto AI API service

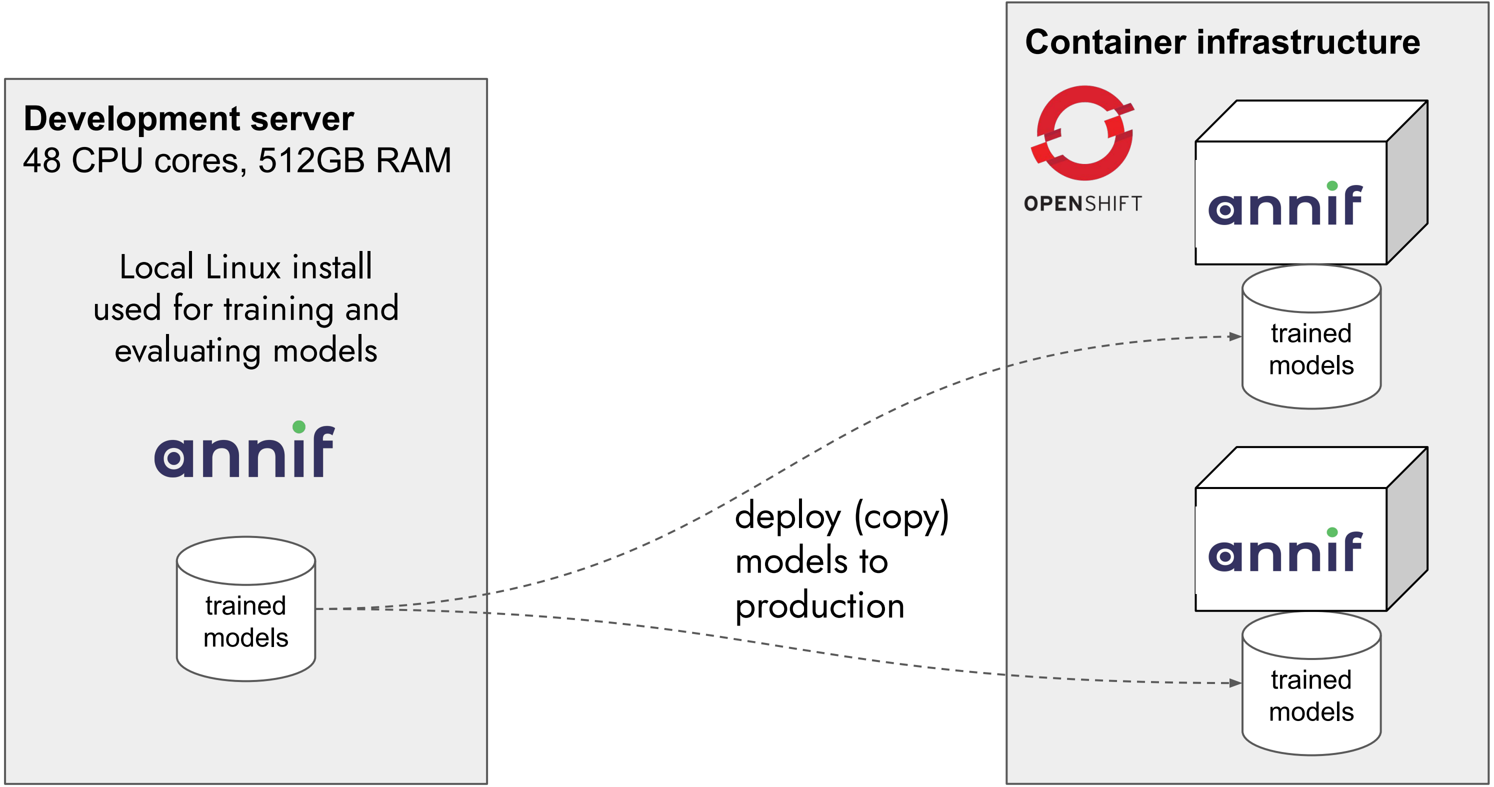


Suggestions	Copy
software development	term URI
machine learning	term URI
computer programmes	term URI
learning	term URI
enterprises	term URI
development (active)	term URI
software engineering	term URI
development projects	term URI
projects	term URI

Copy Finto AI suggestions in Aleph ILS format

Technical infrastructure for production use

You can start with laptops, but production use needs servers!



fintoai

api.annif.org

Start by experimentation, move slowly towards production

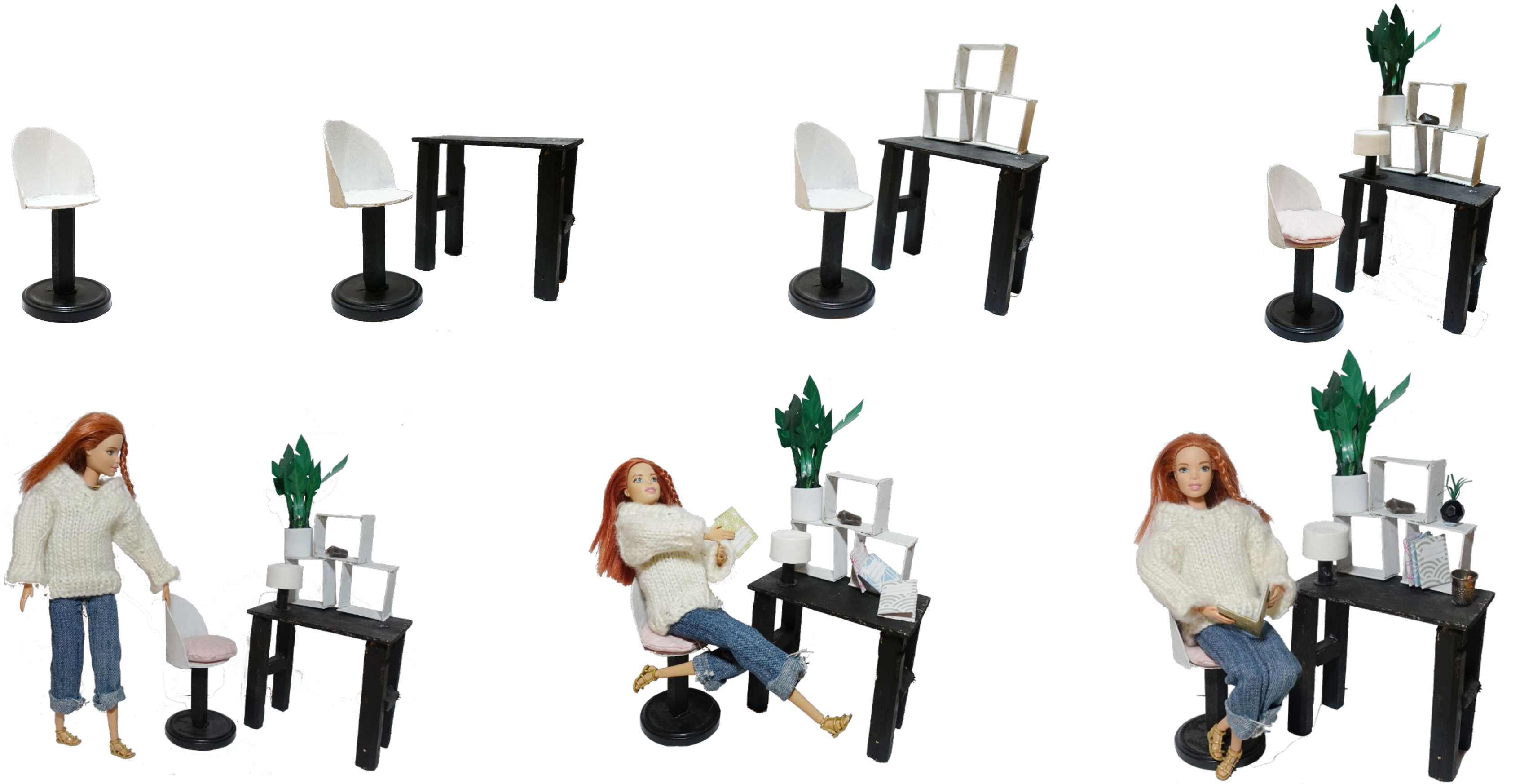


image credit: @kettudolls (IG)

Thank you!

Juho Inkinen

Mona Lehtinen

Osma Suominen

annif.org

osma.suominen@helsinki.fi

Suominen, O., 2019. **Annif: DIY automated subject indexing using multiple algorithms.** *LIBER Quarterly*, 29(1), pp.1–25. DOI: http://doi.org/10.18352/lq.10285

Suominen, O., Inkinen, J., & Lehtinen, M. (2022). **Annif and Finto AI: Developing and Implementing Automated Subject Indexing.** *JLIS.It*, *13*(1), 265–282. https://doi.org/10.4403/jlis.it-12740

These slides: https://tinyurl.com/ifla-supporting-librarians