GENRE ONTOLOGY LEARNING: COMPARING CURATED WITH CROWD-SOURCED ONTOLOGIES

Hendrik Schreiber
tagtraum industries incorporated
hs@tagtraum.com

ABSTRACT

The Semantic Web has made it possible to automatically find meaningful connections between musical pieces which can be used to infer their degree of similarity. Similarity in turn, can be used by recommender systems driving music discovery or playlist generation. One useful facet of knowledge for this purpose are fine-grained genres and their inter-relationships.

In this paper we present a method for learning genre ontologies from crowd-sourced genre labels, exploiting genre co-occurrence rates. Using both lexical and conceptual similarity measures, we show that the quality of such learned ontologies is comparable with manually created ones. In the process, we document properties of current reference genre ontologies, in particular a high degree of disconnectivity. Further, motivated by shortcomings of the established taxonomic precision measure, we define a novel measure for highly disconnected ontologies.

1. INTRODUCTION

In the 15 years since Tim Berners-Lee's article about the *Semantic Web* [2], the *Linking Open Data Community Project* [1] has successfully connected hundreds of datasets, creating a universe of structured data with DBpedia [2] at its center [1, 3]. In this universe, the de facto standard for describing music, artists, the production workflow etc. is *The Music Ontology* [15]. Examples for datasets using it are MusicBrainz/LinkedBrainz [3,4] and DBTune [5]. While in practice taking advantage of *Linked Open Data* (LOD) is not always easy [9], semantic data has been used successfully, e.g. to build recommender systems. Passant et al. outlined how to use LOD to recommend musical content [14]. An implementation of this concept can be found in [13]. Tatlı et al. created a context-based music recommendation system, using genre and instrumentation information from DBpedia [18]. Di Noia et al. proposed a movie recommender based on LOD from DBpedia, Freebase [6], and LinkedMDB [7] [8]. And recently, Oramas et al. created a system for judging artist similarity based on biographies linked to entities in LOD-space [11]. Many of these approaches are trying to solve problems found in recommender systems relying on collaborative filtering, like cold start or popularity bias [4].

Among other data, genre ontologies are a basis for these systems. They allow the determination of degree of similarity for musical pieces (e.g. via the length of the shortest connecting path in the ontology graph), even if we have no other information available. Surprisingly, we know little about the genre ontologies contained in repositories like DBpedia. How large and deep are they? How well do they represent genre knowledge? Are they culturally biased? How interconnected are genres in these ontologies?

While editors of LOD ontologies often follow established rules, it is an inherent property of any ontology that its quality is subjective. An alternative are learned ontologies. Naturally, they do not represent objective truth either, but instead of relying on design principles, they use empirical data. An interesting question is: How do curated genre ontologies compare with learned ontologies?

In the following we are attempting to answer some of these questions. Section 2 starts with proposing a method for building a genre ontology from user-submitted genre tags. In Section 3, we describe the existing genre ontologies DBpedia and WikiData as well as two new ontologies created with the method from Section 2. In Section 4, we describe evaluation measures loaned from the field of ontology learning. Our results are discussed in Section 5, and our conclusions are presented in Section 6.

2. BUILDING THE GENRE GRAPH

As shown in [16], it is possible to create genre taxonomy trees from user-submitted genre labels. These trees have been proven useful for inferring a single top-level genre for a given sub-genre. Unfortunately, taxonomy trees are insufficient when attempting to model the complex inter-genre relations found in the real world. The concept of a fusion-genre for example, i.e. a connection between two

[1] http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData
[2] http://wiki.dbpedia.org/
[3] http://musicbrainz.org/
[4] http://linkedbrainz.org/
[5] http://dbtune.org/
[6] http://www.freebase.com/ — to be shut-down soon.
[7] http://www.linkedmdb.org/

otherwise separate taxonomy trees, is impossible to represent. Therefore, an ontology is commonly regarded as the preferred structure to model genres and their relations.

Similar to [5], we define a genre ontology as a structure $\mathcal{O} = (\mathcal{C}, root, \leq_{\mathcal{C}})$ consisting of a set of concepts $\mathcal{C}$, a designated $root$ concept and the partial order $\leq_{\mathcal{C}}$ on $\mathcal{C} \cup \{root\}$. This partial order is called concept hierarchy. The equation $\forall c \in \mathcal{C} : c \leq_{\mathcal{C}} root$ holds for this concept hierarchy. For the sake of simplicity, we treat the relation between genre names and genre concepts as a bijection, i.e. we assume that each genre name corresponds to exactly one genre concept and vice versa.

To construct a genre ontology based on suitably normalized labels, we first create a genre co-occurrence matrix M as described in [16]. The set $\mathcal{C} = \{c_1, c_2, ..., c_n\}$ contains n genres. Each user submission is represented by a sparse vector $u \in \mathbb{N}^n$ with

$$u_i = \begin{cases} 1, & \text{if } c_i = \text{user-submitted genre} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Each song is represented by a vector $s \in \mathbb{R}^n$. Each s is defined as the arithmetic mean of all user submissions u associated with a given song. Thus s_i describes the relative strength of genre c_i. Co-occurrence rates for a given genre c_i with all other genres can be computed by element-wise averaging all s for which $s_i \neq 0$ is true:

$$M_i = \bar{s}, \quad \forall s \text{ with } s_i \neq 0; M \in \mathbb{R}^{n \times n} \tag{2}$$

Unlike [16], we normalize the co-occurrence rates from M so that the maximum co-occurrence rate of one genre with another is 1. This normalized co-occurrence matrix is called N. Just like M, N is asymmetric. For example, `alternative` strongly co-occurs with `rock`, but `rock` co-occurs not as strongly with `alternative`. We take advantage of this by defining a rule that helps us find sub-genres: If a genre c_i co-occurs with another genre c_j more than a minimum threshold τ, c_j co-occurs with c_i more than a minimum threshold υ, and c_i co-occurs with c_j more than the other way around, then we assume that c_i is a sub-genre of c_j. More formally:

$$\begin{aligned} &\forall c_i, c_j \in \mathcal{C} : c_i <_{\mathcal{C}} c_j \text{ iff} \\ &c_i \neq c_j \wedge N_{i,j} > \tau \wedge N_{j,i} > \upsilon \wedge N_{i,j} > N_{j,i} \end{aligned} \tag{3}$$

Note, that this rule allows one genre to be the sub-genre of multiple other genres. τ controls the co-occurrence rate it takes to be recognized as sub-genre. A low τ leads to more sub-genres and fewer top-level genres. υ ensures that the relationship is not entirely one-sided. As an extreme example, a negative υ would require no co-occurrence of genre c_j with c_i, but c_i could still be a sub-genre of c_j.

Applying (3) makes it easy to find top-level genres, but the resulting hierarchy is rather flat. If a genre is more than one node away from $root$, the rule does not perform well, when it comes to deciding whether a genre is either a sub-genre or a sibling. The reason lies in the fixed parameters τ and υ, which are suitably chosen to find top-level genres, but not sub-genres two or more levels deep. To better determine deep sub-genre relationships starting from a given top-level genre, we apply (3) recursively on each hierarchical sub-structure. So if $\mathcal{C}' \subset \mathcal{C}$ is the set of sub-genres for a $c_k \in \mathcal{C}$, then the co-occurrence matrix N' for $\mathcal{C}'$ can be computed just like N. Because N' is normalized, the same τ and υ are suitable to find $\mathcal{C}'$'s top-level genres, i.e. c_k's direct children. Recursion stops, when the sub-structure consists of at most one genre.

3. ONTOLOGIES

In order to evaluate learned ontologies, we need at least one ontology that serves as reference. This is different from a ground truth, as it is well known that a single truth does not exist for ontologies: Different people create different ontologies, when asked to model the same domain [6, 10, 12, 17]. We chose DBpedia and WikiData as references, which are described in Sections 3.1 and 3.2. Using the definitions and rules from Section 2, we constructed two ontologies. One based on submissions by English speaking users and another based on submissions by international users. They are described in Sections 3.3 and 3.4.

3.1 DBpedia Genre Ontology

DBpedia is the suggested genre extension for *The Music Ontology* and therefore a natural choice for a reference ontology. [8] The part of DBpedia related to musical genres is created by extracting Wikipedia's genre infoboxes. For this to succeed, the DBpedia creation process requires that such infoboxes exist, and that there is a defined mapping from localized infobox to ontology properties. Informally we found that for the English edition of Wikipedia both conditions are usually met. This is not always true for other language editions, e.g. German.

Wikipedia's guidelines [9] define three possible hierarchical relations between genres:

- *Sub-genre*: `heavy metal` < `thrash metal`, `black metal`, `death metal`, etc.
- *Fusion*: `industrial` < `industrial metal` ∧ `heavy metal` < `industrial metal`.
- *Derivative*: `post punk` < `house`, `alternative rock`, `dark wave`, etc.

The *derivative* relation differs from *sub-genre* and *fusion* in that derivative genres are considered "separate or developed enough musicologically to be considered parent/root genres in their own right". As the relation does not fit the general concept of sub-genre or sub-class, we excluded it when building the ontology. Further, we were unable to find a formal definition for the DBpedia relation *stylistic origin*. Based on sample data we interpreted it as the inverse of *derivative*. As such it was also excluded. While this made sense for most genres, it did not for some. The `hip hop` infobox for example, lists `East`

[8] As source for this work, we used DBpedia Live, `http://live.dbpedia.org`.

[9] `https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Music/Music_genres_task_force/Guidelines#Genrebox`

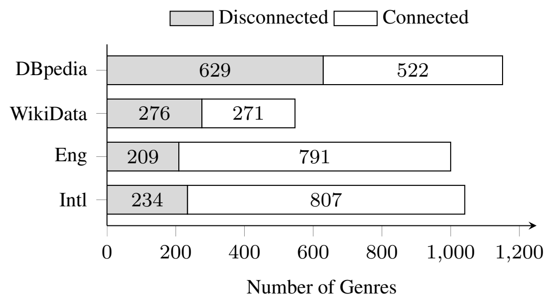

Figure 1. Connected vs. disconnected genres in the four used ontologies. Parameters for generated ontologies: $\tau = 0.17$, $\upsilon = 0.0001$, $|\mathcal{C}_{\text{Eng}}| = 1000$, $|\mathcal{C}_{\text{Intl}}| = 1041$.

`Coast hip hop` and `West Coast hip hop` as regional scenes, but not as sub-genres or derivatives. Unfortunately, in DBpedia, *regional scene* is not defined as a special genre relation, like sub-genre, but just as a plain property. In contrast, both Wikipedia articles on `East Coast hip hop` and `West Coast hip hop` start with assuring a sub-genre relationship to `hip hop`. Also, both DBpedia entries list `hip hop` as the stylistic origin. We found similar issues with `techno` and `Detroit techno`, and other genres.

At the time of writing, the DBpedia-based ontology, created as described above, consisted of 1151 genres with a maximum hierarchy depth of 6. 629 genres (54.6%) did not have any super- or sub-genres (Figure 1). We will refer to it as $\mathcal{O}_{\text{DBpedia}}$. In order to increase the chances of finding corresponding genres in other ontologies, we normalized the raw genre names as well as their aliases found via DBpedia *wikiPageRedirects* (Wikipedia entries for conceptually identical terms).

Loaning from graph theory, we call genres without super- or sub-genres *disconnected*. Ontologies consisting exclusively of disconnected genres we call *trivial*.

3.2 WikiData Genre Ontology

Unlike DBpedia, WikiData is not a parsed version of Wikipedia, but an independent database of structured data for anyone to edit. Currently, WikiData defines just one relation between musical genres: *sub-class*.

In an informal evaluation, we found that, with regard to genres, WikiData is still evolving. While `East Coast hip hop` for example is listed as a sub-genre of `hip hop`, `West Coast hip hop` had no parent at the time of writing. Another example is `techno` and `Detroit techno`. `Detroit techno` existed as an entity, but was not of type music genre, and `techno` was not connected to it in any way. On the plus side, translations of genre names are easily accessible via localized labels for each genre. For matching we used normalized versions of these labels.

At the time of writing, the WikiData-based genre ontology consisted of 547 genres, 276 (50.5%) genres were disconnected, and the hierarchy-depth was 5. We will refer to this ontology as $\mathcal{O}_{\text{WikiData}}$.

3.3 English Language Ontology

Using the rules defined in Section 2, we constructed an ontology based on the top n genre labels submitted by users to the central database of beaTunes [10], a consumer music application [16]. Given the relevance of English in Western pop-culture and the fact that our reference $\mathcal{O}_{\text{DBpedia}}$ offers data based on the English edition of Wikipedia, we only considered submissions by users with English as their system language. We will refer to this ontology as $\mathcal{O}_{\text{Eng}}$. Naturally, $\mathcal{O}_{\text{Eng}}$ is strongly biased towards English culture and contains English genre names almost exclusively. Also, as it is generated from user submitted labels, it contains noise.

Using $\tau = 0.17$ and $\upsilon = 0.0001$ for the top 1000 English genres, we found 209 (20.9%) disconnected genres and the maximum hierarchy-depth was 4.

Because we mentioned `hip hop` and `techno` as problematic examples before, here is what we found for $\mathcal{O}_{\text{Eng}}$: While neither `East Coast hip hop` nor `West Coast hip hop` occur in the top 1000 English genres, `East Coast rap` and `West Coast rap` do. They both have `rap` as a parent, which in turn is a child of `hip hop`. `Techno` does occur as genre, but `Detroit techno` is not in the top 1000 (rank 1557). When using the top 1600 genres as source, `Detroit techno` has `techno` and `electronica` as parents.

3.4 International Ontology

In addition to $\mathcal{O}_{\text{Eng}}$, we generated an international ontology named $\mathcal{O}_{\text{Intl}}$ based on submissions by users with the system languages French, German, Spanish, Dutch, or English. These are the five languages with the most submissions in the beaTunes database. The ontology was created with the goal of being less anglocentric.

Because, the database contains different numbers of submissions per language, we normalized each submission's weight on a per language basis to ensure equal influence. To represent the chosen languages in the selection of the most used genres, we used the intersection of the top n language-specific genres. For $n = 400$ this resulted in a set of 1041 genres, 534 of which also occur in the English top 1000 genres. The sub-set of non-English genre names mostly consists of genuinely new additions like `Kölsch` and `Deutsch Pop`, and translations like `Kindermuziek` and `psychedelische Rock`. The situation regarding `hip hop` and `techno` is similar to $\mathcal{O}_{\text{Eng}}$. Using $\tau = 0.17$ and $\upsilon = 0.0001$ we found that 234 (22.5%) genres were disconnected and the maximum hierarchy-depth was 5.

4. EVALUATION MEASURES

Ontologies can be compared on different levels. In the following, we are going to concentrate on lexical (Section 4.1) and conceptual (Section 4.2) aspects. For both viewpoints measures have been established in the ontology learning community (see e.g. [7, 19]).

[10] `http://www.beatunes.com/`

4.1 Lexical Measures

Let $\mathcal{O}_{\mathrm{R}}$ denote a reference ontology, and $\mathcal{O}_{\mathrm{C}}$ an ontology we wish to evaluate. Correspondingly, $\mathcal{C}_{\mathrm{R}}$ is the set of concepts contained in $\mathcal{O}_{\mathrm{R}}$, and $\mathcal{C}_{\mathrm{C}}$ the concepts in $\mathcal{O}_{\mathrm{C}}$. As we assume a bijective relation between lexical terms and concepts, *lexical precision* (LP) is defined as the ratio between the number of concepts in both ontologies and the number of concepts in $\mathcal{O}_{\mathrm{C}}$:

$$LP(\mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) = \frac{|\mathcal{C}_{\mathrm{C}} \cap \mathcal{C}_{\mathrm{R}}|}{|\mathcal{C}_{\mathrm{C}}|} \tag{4}$$

Lexical recall (LR) is defined as the ratio between the number of concepts in both ontologies and the number of concepts in $\mathcal{O}_{\mathrm{R}}$ [5]:

$$LR(\mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) = \frac{|\mathcal{C}_{\mathrm{C}} \cap \mathcal{C}_{\mathrm{R}}|}{|\mathcal{C}_{\mathrm{R}}|} \tag{5}$$

Finally, the *lexical F-measure* (LF) is defined by:

$$LF(\mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) = \frac{2 \cdot LP \cdot LR}{LP + LR} \tag{6}$$

4.2 Conceptual Measures

The similarity of two concepts $c_i \in \mathcal{C}_{\mathrm{C}}$ and $c_j \in \mathcal{C}_{\mathrm{R}}$ can be measured by comparing their *semantic cotopies* [10]. A basic semantic cotopy is defined as the set containing all super- and sub-concepts for a given concept including itself. The *common semantic cotopy* (*csc*) is similar, but only takes concepts into account that are members of both ontologies we wish to compare. Additionally, the concept for which we are building the cotopy is excluded ($<_{\mathcal{C}}$ instead of $\leq_{\mathcal{C}}$). Both modifications are intended to minimize the influence of lexical similarity [5]:

$$\begin{aligned}&\mathrm{csc}(c_i, \mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) \\ &\quad = \{c_j \in \mathcal{C}_{\mathrm{C}} \cap \mathcal{C}_{\mathrm{R}} | c_j <_{\mathcal{C}_{\mathrm{C}}} c_i \vee c_i <_{\mathcal{C}_{\mathrm{C}}} c_j\}\end{aligned} \tag{7}$$

The *local taxonomic precision* (tp_{csc}) is defined as the ratio between the size of the intersection of the cotopies for two concepts, and the size of the cotopy of just the concept to evaluate:

$$\begin{aligned}&tp_{\mathrm{csc}}(c_i, c_j, \mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) \\ &\quad = \frac{|\mathrm{csc}(c_i, \mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) \cap \mathrm{csc}(c_j, \mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}})|}{|\mathrm{csc}(c_i, \mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}})|}\end{aligned} \tag{8}$$

tp_{csc} is undefined for $|\mathrm{csc}(c_i, \mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}})| = 0$ (division by zero). In the spirit of [5], i.e. to avoid unjustifiably high values for trivial ontologies, we define $tp_{\mathrm{csc}} = 0$ for this case. Based on the local tp_{csc}, we define a *global taxonomic precision* (TP_{csc}) as the mean tp_{csc} for all concepts in $\mathcal{C}_{\mathrm{C}} \cap \mathcal{C}_{\mathrm{R}}$ [7]:

$$\begin{aligned}&TP_{\mathrm{csc}}(\mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) \\ &\quad = \frac{1}{|\mathcal{C}_{\mathrm{C}} \cap \mathcal{C}_{\mathrm{R}}|} \sum_{c \in \mathcal{C}_{\mathrm{C}} \cap \mathcal{C}_{\mathrm{R}}} tp_{\mathrm{csc}}(c, c, \mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}})\end{aligned} \tag{9}$$

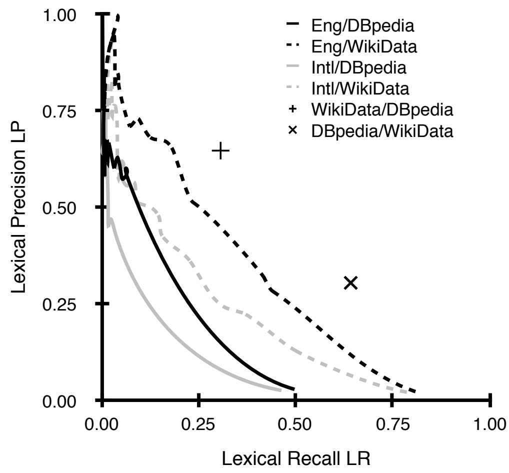

Figure 2. Lexical precision LP and recall LR for learned ontologies $\mathcal{O}_{\mathrm{Eng}}$ and $\mathcal{O}_{\mathrm{Intl}}$ based on different genre numbers. $\mathcal{O}_{\mathrm{DBpedia}}$ and $\mathcal{O}_{\mathrm{WikiData}}$ serve as reference ontologies (with a fixed number of genres).

$\mathcal{O}_{\mathrm{C}}$	$\mathcal{O}_{\mathrm{WikiData}}$	$\mathcal{O}_{\mathrm{Eng}}$	$\mathcal{O}_{\mathrm{Intl}}$
LP	0.644	0.260	0.183
LR	0.306	0.226	0.166
LF	0.415	0.242	0.174
TP_{csc}	0.098	0.187	0.193
TR_{csc}	0.114	0.220	0.212
TF_{csc}	0.105	0.202	0.202
TP_{con}	0.266	0.237	0.240
TR_{con}	0.319	0.278	0.268
TF_{con}	0.290	0.256	0.253

Table 1. Results for $\mathcal{O}_{\mathrm{R}} = \mathcal{O}_{\mathrm{DBpedia}}$, $\tau = 0.17$, $\upsilon = 0.0001$, $|\mathcal{C}_{\mathrm{Eng}}| = 1000$, $|\mathcal{C}_{\mathrm{Intl}}| = 1041$.

The *taxonomic recall* (TR_{csc}) is:

$$TR_{\mathrm{csc}}(\mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) = TP_{\mathrm{csc}}(\mathcal{O}_{\mathrm{R}}, \mathcal{O}_{\mathrm{C}}) \tag{10}$$

Finally, the *taxonomic F-measure* (TF_{csc}) is defined by:

$$TF_{\mathrm{csc}}(\mathcal{O}_{\mathrm{C}}, \mathcal{O}_{\mathrm{R}}) = \frac{2 \cdot TP_{\mathrm{csc}} \cdot TR_{\mathrm{csc}}}{TP_{\mathrm{csc}} + TR_{\mathrm{csc}}} \tag{11}$$

5. RESULTS AND DISCUSSION

We measured the similarity of all four ontologies using varying parameters for the learned ones. Section 5.1 reports lexical results, Section 5.2 conceptual results. In Section 5.3 we discuss our findings.

5.1 Lexical Results

How similar are the ontologies on the lexical level? For the reference ontologies $\mathcal{O}_{\mathrm{DBpedia}}$ and $\mathcal{O}_{\mathrm{WikiData}}$ this is easy to answer: $LP/LR/LF(\mathcal{O}_{\mathrm{WikiData}}, \mathcal{O}_{\mathrm{DBpedia}}) = 0.64/0.31/0.42$ (Table 1). Given their respective sizes, the highest possible values for this pairing are $1.00/0.48/0.64$ (if $\mathcal{C}_{\mathrm{WikiData}} \subset \mathcal{C}_{\mathrm{DBpedia}}$).

$\mathcal{O}_{\text{C}}$	$\mathcal{O}_{\text{DBpedia}}$	$\mathcal{O}_{\text{Eng}}$	$\mathcal{O}_{\text{Intl}}$
LP	0.303	0.259	0.202
LR	0.638	0.473	0.384
LF	0.411	0.335	0.264
TP_{csc}	0.114	0.174	0.181
TR_{csc}	0.098	0.151	0.149
TF_{csc}	0.105	0.162	0.163
TP_{con}	0.319	0.305	0.357
TR_{con}	0.266	0.274	0.303
TF_{con}	0.290	0.288	0.328

Table 2. Results for $\mathcal{O}_{\text{R}}$= $\mathcal{O}_{\text{WikiData}}$, $\tau = 0.17$, $\upsilon = 0.0001$, $|\mathcal{C}_{\text{Eng}}|= 1000$, $|\mathcal{C}_{\text{Intl}}|= 1041$.

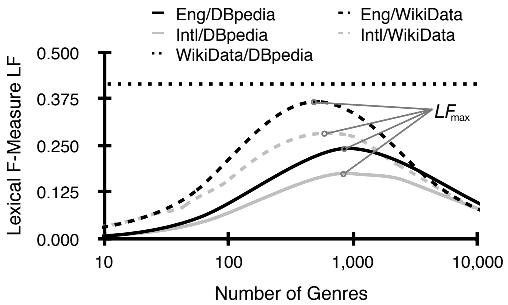

Figure 3. Lexical F-measure LF for learned ontologies $\mathcal{O}_{\text{Eng}}$ and $\mathcal{O}_{\text{Intl}}$ based on different genre numbers. $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$ serve as reference ontologies (with a fixed number of genres).

For the learned ontologies, the answer depends on the number of genres used during generation. Not surprisingly, we observed that recall increases with the number of genres, while precision decreases. When comparing precision/recall values for the learned ontologies with $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$, values for $\mathcal{O}_{\text{WikiData}}$ are predominantly higher, indicating a greater similarity with the learned ontologies (dashed lines in Figure 2). This is also reflected in the lexical F-measure shown in Figure 3. While $LF_{\text{max}}(\mathcal{O}_{\text{Eng}}, \mathcal{O}_{\text{DBpedia}})$ is only 0.24, $LF_{\text{max}}(\mathcal{O}_{\text{Eng}}, \mathcal{O}_{\text{WikiData}})$ is 0.37—just 0.05 below $LF(\mathcal{O}_{\text{WikiData}}, \mathcal{O}_{\text{DBpedia}})$, shown as dotted line. For $\mathcal{O}_{\text{Intl}}$, the LF_{max} values are lower than their $\mathcal{O}_{\text{Eng}}$ counterparts: $LF_{\text{max}}(\mathcal{O}_{\text{Intl}}, \mathcal{O}_{\text{DBpedia}})$ is 0.18 and $LF_{\text{max}}(\mathcal{O}_{\text{Intl}}, \mathcal{O}_{\text{WikiData}})$ is 0.28. In all cases, the number of genres needed to achieve LF_{max} approximately equals the number of genres in the reference ontology.

When generated for very few genres, both learned ontologies reach $LP = 1.0$ for either reference ontology, as they all contain the top genres `rock`, `pop`, etc. The achievable LR values however, differ significantly. At a very low precision level, both learned ontologies reach no more than $LR = 0.5$ with $\mathcal{O}_{\text{DBpedia}}$ as reference. In contrast, at the same precision level, with $\mathcal{O}_{\text{WikiData}}$ as reference, LR is greater than 0.74 (Figure 2). We investigated what might be the reason for the low recall for $\mathcal{O}_{\text{DBpedia}}$ and came to the conclusion that it contains many genres that are unlikely to be found in standard genre tags, e.g. `Music of Innsbruck` or `Music of Guangxi`.

5.2 Conceptual Results

Just like the lexical results, conceptual results depend on the number of genres considered and of course the reference used. Additionally, τ and υ influence the outcome.

We found that values for $\upsilon \leq 0.0001$ hardly affect $TP/TR/TF$ results, when the learned ontology is compared with $\mathcal{O}_{\text{DBpedia}}$ or $\mathcal{O}_{\text{WikiData}}$. However, inspection of the learned ontologies shows, that a very low υ causes some genres to have significantly more parents than the average genre. Consequently, they connect unrelated parts of the ontology. Examples for this are `canadian` and `seventies`. We argue that neither is really a musical genre, but rather an orthogonal concept—a region and an era, respectively. This also explains why $TP/TR/TF$ are unaffected, as by definition they are only influenced by genres that appear in both the learned and the reference ontology. Being orthogonal to the genre concept, they never occur in a reference ontology. We further observed, that υ values greater than 0.0001 affect $TP/TR/TF$ negatively. The following data are therefore based on $\upsilon = 0.0001$.

We investigated how τ influences $TP/TR/TF$ by calculating TF_{csc} for $\mathcal{O}_{\text{Eng}}$ ($|\mathcal{C}_{\text{Eng}}|$=1000) and $\mathcal{O}_{\text{Intl}}$ ($|\mathcal{C}_{\text{Intl}}|$=1041) with $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$ as reference ontologies. Based on Figure 4, we chose $\tau = 0.17$ as a value reasonably suited for all ontologies.

Keeping τ and υ constant, how are taxonomic results influenced by the number of genres? $TF_{\text{csc}}(\mathcal{O}_{\text{Eng}}, \mathcal{O}_{\text{DBpedia}})$ peaks around 160 genres with $TF_{\text{max}} = 0.31$. The same is true for $TF_{\text{csc}}(\mathcal{O}_{\text{Eng}}, \mathcal{O}_{\text{WikiData}})$ with $TF_{\text{max}} = 0.32$. For $TF_{\text{csc}}(\mathcal{O}_{\text{Intl}}, \mathcal{O}_{\text{DBpedia}})$ we found TF_{max} around 285 genres with a value of 0.26 and for $TF_{\text{csc}}(\mathcal{O}_{\text{Intl}}, \mathcal{O}_{\text{WikiData}})$ around 411 genres with 0.28 (Figure 5a). In all cases, TF_{csc} peaks for genre numbers that are well below the number of genres in the reference ontology. This makes sense as all ontologies, to a large degree, consist of disconnected genres that cannot contribute to a higher TF_{csc}. But even for most non-TF_{max} genre numbers, TF_{csc} values involving the learned ontologies are higher than $TF_{\text{csc}}(\mathcal{O}_{\text{WikiData}}, \mathcal{O}_{\text{DBpedia}}) = 0.12$, depicted as the dotted line in Figure 5a. It appears, as if both $\mathcal{O}_{\text{Eng}}$ and $\mathcal{O}_{\text{Intl}}$ are taxonomically more similar to $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$ than $\mathcal{O}_{\text{DBpedia}}$ to $\mathcal{O}_{\text{WikiData}}$ or the other way around. Upon closer inspection, we attributed this to the greater intersection of disconnected genres from $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$. 47.6% of the genres in the lexical intersection $\mathcal{C}_{\text{DBpedia}} \cap \mathcal{C}_{\text{WikiData}}$ are disconnected in at least one of the two ontologies. But only 36.9% of the $\mathcal{O}_{\text{WikiData}}$ intersection with $\mathcal{O}_{\text{Eng}}$ and 38.6% of the intersection with $\mathcal{O}_{\text{Intl}}$ are disconnected. Even lower are the values for $\mathcal{O}_{\text{DBpedia}}$: Just 17.7% of the intersection with $\mathcal{O}_{\text{Eng}}$ and 16.7% of the intersection with $\mathcal{O}_{\text{Intl}}$ are disconnected. As defined in Section 4.2, disconnected genres lead to zero tp_{csc} values. This significantly contributes to $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$ achieving lower

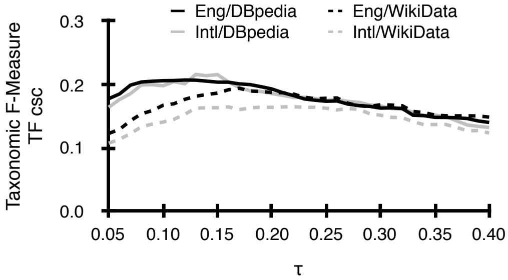

Figure 4. Taxonomic F-measure TF_{csc} for $\mathcal{O}_{\text{Eng}}$ ($|\mathcal{C}_{\text{Eng}}|$=1000) and $\mathcal{O}_{\text{Intl}}$ ($|\mathcal{C}_{\text{Intl}}|$=1040) compared with $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$ for varying τ values.

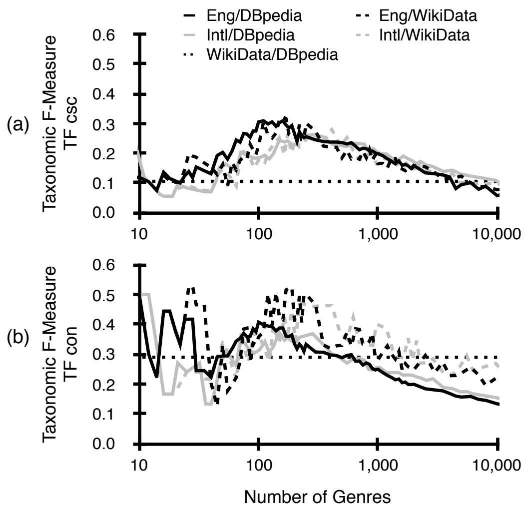

Figure 5. Taxonomic F-measures TF_{csc} and TF_{con} for learned ontologies and different genre numbers compared with $\mathcal{O}_{\text{DBpedia}}$ and $\mathcal{O}_{\text{WikiData}}$. $\tau = 0.17$, $\upsilon = 0.0001$.

$TP/TR/TF$ values, when compared with each other, than a pairing that does not have as many disconnected genres in common. By removing all disconnected genres from $\mathcal{C}_{\text{C}} \cap \mathcal{C}_{\text{R}}$ before calculating TP_{csc}, we calculated the *connected taxonomic precision* (TP_{con}), which results in higher values for all pairings, and especially for ($\mathcal{O}_{\text{DBpedia}}$,$\mathcal{O}_{\text{WikiData}}$) (Figure 5b). The problem with genre ontologies is, that from a taxonomic point of view, the reference ontologies are, to a large degree, trivial. TP_{con} attempts to work around the problem by accepting that there are disconnected genres and ignores them when calculating TP.

5.3 Discussion

The results show that, using the proposed method, it is possible to create an ontology that is almost as similar to $\mathcal{O}_{\text{WikiData}}$ as the alternative reference ontology $\mathcal{O}_{\text{DBpedia}}$—on both the lexical and conceptual level. When comparing learned ontologies with the more comprehensive $\mathcal{O}_{\text{DBpedia}}$, the results are not quite as good: while it is possible to generate an ontology that is as similar to $\mathcal{O}_{\text{DBpedia}}$ as $\mathcal{O}_{\text{WikiData}}$ on the conceptual level, it was not possible on the lexical level due to the many uncommon genres contained in DBpedia.

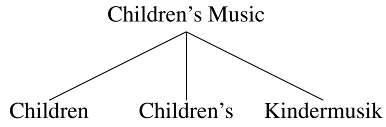

Figure 6. Declinations and translations in $\mathcal{O}_{\text{Intl}}$.

Sourcing genre tags from international instead of just English users has proven detrimental to lexical similarity, when comparing with either $\mathcal{O}_{\text{DBpedia}}$ or $\mathcal{O}_{\text{WikiData}}$. When inspecting $\mathcal{O}_{\text{Intl}}$, we noted translations and declinations of genre names. They are often close relatives in the generated hierarchy (e.g. Figure 6). On one hand, this clearly contributed to worse lexical results. On the other hand, we see this as a potentially useful property. Different crowd-sourced notations in a reference ontology simplify lookups, because there is no mismatch between the names that are really being used and the names that occur in the ontology. Furthermore, it allows easy measurement of semantic similarity for unknown notations or translations, e.g. via the length of the shortest connecting path. It also adds a cultural dimension, as `children's music` and `Kindermusik` are clearly the same genre, but a parent looking for music may prefer music from its own culture and chooses one genre over the other.

All differences put aside, one must not forget that the mentioned ontologies can be linked and thus complement each other. A missing connection in one ontology, may be made through another one. The generated ontologies can be found at `http://www.tagtraum.com/learned_ontologies.html` and contain *sameAs*-relations with WikiData and DBpedia.

6. CONCLUSION AND FUTURE WORK

DBpedia and WikiData both consist of two parts: The first part contains disconnected genres that have neither parents nor sub-genres. It has little value in a taxonomic sense, but can still serve as linkable data in LOD-space. The second part is an imperfect, but rich, interconnected hierarchy of relatively popular genres that can be used for similarity estimation and therefore recommender systems. Because of the way DBpedia is created, not all language editions are represented equally well.

By exploiting co-occurrence rates of user submitted genre labels, we were able to learn new genre ontologies. Using established lexical and conceptual similarity measures, we successfully demonstrated the validity of the proposed learning method. Further, to improve conceptual similarity measures with largely trivial reference ontologies, we proposed an additional measure, the connected taxonomic precision.

Future work may add translation recognition and improve genre name normalization. Taking advantage of learned genre ontologies may lead to interesting new music information retrieval applications.

7. REFERENCES

[1] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. *DBpedia: A nucleus for a web of open data*. Springer, 2007.

[2] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 284(5):28–37, 2001.

[3] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia–a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165, 2009.

[4] Òscar Celma and Pedro Cano. From hits to niches? or how popular artists can bias music recommendation and discovery. In *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*, page 5. ACM, 2008.

[5] Philipp Cimiano, Andreas Hotho, and Steffen Staab. Learning concept hierarchies from text corpora using formal concept analysis. *Journal of AI Research (JAIR)*, 24:305–339, 2005.

[6] Philipp Cimiano, Alexander Mädche, Steffen Staab, and Johanna Völker. Ontology learning. In *Handbook on ontologies*, pages 245–267. Springer, 2009.

[7] Klaas Dellschaft and Steffen Staab. On how to perform a gold standard based evaluation of ontology learning. In *The Semantic Web-ISWC 2006*, pages 228–241. Springer, 2006.

[8] Tommaso Di Noia, Roberto Mirizzi, Vito Claudio Ostuni, Davide Romito, and Markus Zanker. Linked open data to support content-based recommender systems. In *Proceedings of the 8th International Conference on Semantic Systems*, pages 1–8. ACM, 2012.

[9] Ben Fields, Sam Phippen, and Brad Cohen. A case study in pragmatism: exploring the practical failure modes of linked data as applied to classical music catalogues. In *Proceedings of the 2nd International Workshop on Digital Libraries for Musicology*, pages 21–24. ACM, 2015.

[10] Alexander Mädche and Steffen Staab. Measuring similarity between ontologies. In *Knowledge engineering and knowledge management: Ontologies and the semantic web*, pages 251–263. Springer, 2002.

[11] Sergio Oramas, Mohamed Sordo, Luis Espinosa-Anke, and Xavier Serra. A semantic-based approach for artist similarity. In *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, pages 100–106, Málaga, Spain, 2015.

[12] François Pachet and Daniel Cazaly. A taxonomy of musical genres. In *Proceedings of the Content-based Multimedia Information Access Conference (RIAO)*, pages 1238–1245, Paris, France, 2000.

[13] Alexandre Passant. dbrec–music recommendations using DBpedia. In *The Semantic Web–ISWC 2010*, pages 209–224. Springer, 2010.

[14] Alexandre Passant and Yves Raimond. Combining social music and semantic web for music-related recommender systems. In *The 7th International Semantic Web Conference*, volume 19, 2008.

[15] Yves Raimond, Samer Abdallah, Mark Sandler, and Frederick Giasson. The music ontology. In *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, pages 417–422, Vienna, Austria, 2007.

[16] Hendrik Schreiber. Improving genre annotations for the million song dataset. In *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, pages 241–247, Málaga, Spain, 2015.

[17] Mohamed Sordo, Òscar Celma, Martín Blech, and Enric Guaus. The quest for musical genres: Do the experts and the wisdom of crowds agree? In *Proceedings of the 9th International Society for Music Information Retrieval Conference (ISMIR)*, pages 255–260, 2008.

[18] İpek Tatlı and Ayşenur Birtürk. Using semantic relations in context-based music recommendations. In *Workshop on Music Recommendation and Discovery (WOMRAD), ACM Conference on Recommender Systems.*, pages 14–17, Chicago, IL, USA, 2011.

[19] Wilson Wong, Wei Liu, and Mohammed Bennamoun. Ontology learning from text: A look back and into the future. *ACM Computing Surveys (CSUR)*, 44(4):20, 2012.