ELMDist: A vector space model with words and MusicBrainz entities

Luis Espinosa-Anke[1], Sergio Oramas[2], Horacio Saggion[1], and Xavier Serra[2]

[1] TALN Natural Language Processing Group - Universitat Pompeu Fabra
[2] Music Technology Group - Universitat Pompeu Fabra

Abstract. Music consumption habits as well as the Music market have changed dramatically due to the increasing popularity of digital audio and streaming services. Today, users are closer than ever to a vast number of songs, albums, artists and bands. However, the challenge remains in how to make sense of all the data available in the Music domain, and how current state of the art in Natural Language Processing and semantic technologies can contribute in Music Information Retrieval areas such as music recommendation, artist similarity or automatic playlist generation. In this paper, we present and evaluate a distributional sense-based embeddings model in the music domain, which can be easily used for these tasks, as well as a device for improving artist or album clustering. The model is trained on a disambiguated corpus linked to the MusicBrainz musical Knowledge Base, and following current knowledge-based approaches to sense-level embeddings, entity-related vectors are provided *à la* WordNet, concatenating the id of the entity and its mention. The model is evaluated both intrinsically and extrinsically in a supervised entity typing task, and released for the use and scrutiny of the community.

1 Introduction

One of the earliest avenues for improvement identified in the otherwise powerful word embeddings [1,2] is that they tend to "conflate" (or agglutinate) in one vector the semantic representation of several meanings of a word or phrase [3]. In the last years, however, we have witnessed two parallel directions for alleviating this weakness. On one hand, what we could call *unsupervised approaches*, which usually cluster contexts in which a word appears and then obtain a representation of each cluster [4,5,6]. On the other hand, the so-called *knowledge-based approaches* exploit predefined semantic representations encoded in lexicons or Knowledge Bases (KBs) such as WordNet [7] or BabelNet [8]. Prominent examples include, *inter alia*, [9,10,11,12,13]. While these approaches have shown competitive results in some of the classic tasks in Natural Language Processing (NLP) like semantic similarity, whether these models would be truly helpful in restricted domains of knowledge remains an open question. In fact, they are inherently flawed by the natural incapability of current KBs and semantic lexicons to capture *all the knowledge existing out there.* While this is a problem theoretically addressed by the Open Information Extraction (OIE) paradigm [14,15,16], the truth is that current OIE systems are still too noisy and error-prone, and even approaches that have attempted to integrate them have had to deal with issues related with sparsity, redundancy and the lack of ontologization [17]. Another direction for improving *sense-level* vector representations in specific domains of knowledge is the construction and annotation of large domain corpora, and *transfer* the knowledge acquired from previously published (and highly successful) vector space modeling algorithms to a target domain. This is precisely the direction we adopt in this work.

In this paper, we present ELMDist[3] a *sense-level* embeddings model in the music domain, trained on a music-specific corpus of artist biographies, where musical entities have been automatically annotated with high precision against the musical KB MusicBrainz (MB) [18]. We evaluate this model in a twofold strategy. First, a qualitative evaluation of nearest neighbours to assess artist similarity. And second, a quantitative evaluation, in which we devise an *entity typing* strategy so that, for a given vector, we predict the probability of it being any of four of the most common entities in the music domain, namely `artist`, `album`, `song` and `record label`. Our results show a surprisingly good precision, especially considering the small size of the corpus, while coverage could be assumed to increase as additional corpora are incorporated to the model. We make available for the community a set of disambiguated pretrained vectors, as well as dumps of matrices trained to learn (`artist`, `album` and `record label`)-wise transformations.

2 Method

In this section, we first flesh out the different resources our approach consists of. First, we briefly summarize the approach followed to construct an automatic and fully disambiguated corpus in the music domain (Sect. 2.1). Second, we describe the linear transformation approach followed for assigning a music-specific type to any vector (Sect. 2.2). Finally, we provide evaluation results in Sect. 3.

2.1 Entity Linking in the Music Domain

While there is not a substantial work in applying current state of the art NLP systems in the music domain, this scenario seems to be gradually shifting, especially since exploitation of text mining techniques has proven to be useful for Music Information Retrieval (MIR) tasks such as artist similarity [19] or music recommendation [20,21]. One of the greatest challenges posed by the music domain for text understanding lies on the fact that musical entities show high variability, arguably higher than the regular entities with which evaluation is usually concerned in Entity Recognition tasks, like Person, Location or Organization. Notable examples attempting Entity Linking (EL) (the task to assign

[3] Available at `https://bitbucket.org/luisespinosa/elmdist/`

to an entity mention its corresponding entry or uri in a predefined inventory) include the detection of music-related entities (e.g. songs or bands) on informal text [22] or applying Hidden Markov Models for discovering musical entities in Chinese corpora [23].

In this work, we use as training data an extended version of the ELMD corpus [24], which stems from the collection of biographies acquired from Last.fm[4]. The original ELMD corpus contains annotations mapped between inner last.fm links and their corresponding DBpedia URI thanks to the voting algorithm described in [24]. However, the whole motivation for this work is to model musical entities against a music-specific KB so that the coverage of entities is higher. Hence, we leverage ELMD2[5], an extension of the original annotated corpus, and take advantage of the fact that a large portion of last.fm annotations have a direct mapping to MB via its API. Furthermore, existing annotations in every document are propagated, assuming they appear in a one-sense-per-discourse fashion. For example, if the text span *The Beatles* is marked as an annotation in the first sentence of a document, and it appears again in the second sentence, but there is no annotation associated, an annotation is added. Finally, we look for mentions of the entity that constitutes the main theme of the biography, and annotate all its mentions within the biography, assuming unambiguity. The number of annotations and distinct entities are reported in Tables 1. Note that MB has a coverage of 93.6% over all the annotations, and 91.1% over all distinct entities.

	Annotations	Entities
All	144,593	63,902
Artist	112,524	39,131
Album	18,701	15,064
Song	9,203	7,832
Label	4,165	1,875

Table 1. Statistics of the ELMD2 corpus from which the ELMDist vector space model is derived. Annotations refers to all distinct mentions or apparitions of an entity of its corresponding type, whereas the Entities column refers to the number of distinct entities of each type.

2.2 Training a sense-level embeddings model

Taking advantage of the mapping existing in ELMD between last.fm links and MB ids (mbids), we follow [25,26] and, for each entity mention in the ELMD corpus, we concatenate its mention with its corresponding mbid, so that this

[4] http://last.fm

[5] Described in http://mtg.upf.edu/download/datasets/elmd.

"sense" (in an analogy with WordNet) is assigned one single vector. For instance, given the input sentence:

> *The Tools of the Trade* was never distributed outside the US, and yet again *Nocturnal Breed* would have to look for other business interests.

the resulting disambiguated concepts would result in (note that we also include the *type* of each entity)[6]

> **the_tools_of_the_trade_album_mbid:fb410e8f** was never distributed outside the US, and yet again **nocturnal_breed_artist_mbid:f267a071bb23** would have to look for other business interests.

We use the *gensim*[7] implementation of *word2vec* [27], and train a CBOW model of 300 dimensional vectors, filtering out tokens with a frequency less than 3, with a context window of 5 tokens, and hierarchical softmax (usually a better performing algorithm for infrequent words). Due to the nature of the corpus, we consider each disambiguated entity mention as a single token, and hence assign it a unique vector.

3 Model Evaluation

In this section, we provide the reader with the result of two experiments where we assess the fitness of the model, first, for artist similarity, and second, for named entity typing.

3.1 Entity Similarity

Artist similarity is an important task in MIR. Knowing, for instance, similar artists to the band `ZZ Top` (e.g. bands belonging to the jazz-rock genre), allows for a better music recommendation and playlist suggestion, and ultimately leads to a better user experience. While artist similarity has been approached looking at score, acoustic or even cultural features [28], text-based approaches have also played an important role in this task. For instance, by computing co-occurrences of artist names [29], leveraging search engines result counts [30] or introducing further linguistic analysis in the form of ngram, part of speech and TFIDF information [31].

Manual Evaluation In the first experiment, we asked 2 human judges to assess whether, given an input artist, the disambiguated nearest neighbours in the vector space were *similar*[8]. We randomly sampled 10 instances of each type.

[6] For readability purposes, we have shortened the `mbid` of the annotated entities.

[7] `https://radimrehurek.com/gensim/models/word2vec.html`

[8] Since this judgement is, in the end, a subjective decision, we did not ask them to look at data such as listening habits.

Note that judging whether two songs are similar is much easier than judging whether two record labels are similar, and for the latter, we suggested that the judges looked at whether these record labels had a preference for a certain music genre, or if they were based (or originated) in the same geographical location. Then, for each test instance, we retrieved the top 3 entities returned by cosine distance. This results in 40 evaluation instances for each of the considered music types.

Results, shown in Table 2, show that the model clusters together not only vectors of the same type, but also sharing some kind of relationship, as assessed by the judges. Still, the outcome of this experiment is affected by subjectivity. For instance, given one of the randomly sampled instances for evaluation was the record label `Universal Records`, and evaluators were given as nearest neighbors other record labels which shared some features (e.g. also based in Paris or London), but which had little relationship from the musical standpoint. We show the behaviour of the model in Table 3, where the difference between the quality of *artist* and *record label* vectors can be clearly seen, as opposed to the quality of `songs` and `albums`. We plan to further investigate this notorious discrepancy.

`artist`	`album`	`r.label`
48%	20%	44%

Table 2. Average precision for the entity similarity task

`artist`	`album`
Nirvana	*Heaven and Hell (Black Sabbath)*
Metallica	Shaman (Brazilian Progressive Rock Band)
Kinks	The Boys Next Door (Nick Cave Album)
Tiga	changing
NOFX	shortening
Megadeth	shortened
`song`	`record label`
Stand By Me (Ben E. King)	*London Records*
Gimme Little Sign (Brenton Wood)	Atlantic Records
doble	Epic Records
petite	Merge Records
zur	Elektra Records
rad	Universal Records

Table 3. Examples of well known input entities for each type (in italics), showcasing the type of nearest neighbours that ELMDist provides.

We found surprisingly high results in the `record label` entity, despite the subjective nature of this classification, where almost half of the nearest neighbours were *similar* record labels to the input entity. However, we did encounter (also surprsingly) poor results in the `song` entity, where only 2 out of every 10 cases were deemed similar by the judges. There was an average observed agreement of 80% between both judges. Finally, in *all* cases the nearest neighbours to the validation `album` vectors we used were songs, and since we asked our judges to only consider similarity for entities of the same type (i.e., for albums, only albums), results for this entity type are not reported.

Automatic Evaluation The literature in distributional semantics has in general explored whether *co-hyponymy*, the property of sharing the same hypernym, can be considered as a measure or indicator of similarity. Since the ELMDist vectors include type information, we conduct a similar experiment as the manual evaluation described in Section 3.1. In this case, for all vectors of any of the available types, we collect their five closest vectors by cosine score and assess their similarity. Without human intervention, we follow a coarse-grained classification in which we consider similar two vectors sharing the same type. We report two types of precision-wise evaluation: first, considering all neighbouring vectors as candidates (all vecs.), and second, considering only the disambiguated vectors (disambig.) as candidates (Table 4). In both cases, it is interesting to note the better performance of this approach for the `artist` and `record label` types. This is likely due to the fact that, in the former case, there are many more artist entities and therefore there is extensive corpus-based evidence in the form of word context to generate reliable representations. In the latter case, however, the better performance seems to be due to the less variability in how record labels are referred to in last.fm biographies, with less linguistic variability and therefore *better* vectors with substantially less data. Further discussion about the case of the record label type is provided in Section 3.2.

	all vecs.	**disambig.**
artist	0.53	0.96
album	0.19	0.41
song	0.13	0.38
label	0.31	0.62

Table 4. Automatic evaluation of the similarity experiment considering co-hyponymy as criterion.

3.2 Named Entity Typing

Hypernymy is an important semantic relation that has to be accounted for in automatic text understanding. For instance, knowing that Tom Cruise is an actor can help a question answering system answer the question “which actors are involved in Scientology?” [32]. Similarly, in the music domain it is important to detect mentions of music entities such as bands or albums. This can be useful for automatically inserting new entries in existing KBs, or for improving any of the MIR tasks we have mentioned earlier.

We thus proceed to evaluate our model in the task of automatic entity *typing* (restricting the number of available types to ARTIST, ALBUM, SONG and RECORD LABEL). The task consists in, given a *text-level* (non-disambiguated) input entity, predict its most likely musical type. To this end, we follow [33], who showed that semantically related pairs of linguistic items (x, y) could be modeled in terms of a linear transformation between them, having both items existing in two different analogous spaces. The original work by [33] used this intuition for modeling a transformation between English and Spanish (i.e., for word-level machine translation). This has been further explored for constructing semantic hierarchies in Chinese [34], Twitter language normalization [35], or for collocation discovery [36].

We follow this line of research, and construct an *entity matrix* $\mathbf{E} = [\boldsymbol{x}_1 \ldots \boldsymbol{x}_n]$ and a *music type matrix* $\mathbf{T} = [\boldsymbol{y}_1 \ldots \boldsymbol{y}_n]$, where $\mathbf{E}$ is our newly trained model, and $\mathbf{T}$ is the pretrained word2vec vectors on the Google News corpus[9]. These matrices are constructed as follows. We randomly sample musical entities from our musical model, and depending on their type (field *category* in the annotated corpus), we assign them a *set of prototypical words* for each type[10]. For instance, if we found the album *Nevermind* (by Nirvana), we would train with pairs such as ($\textit{Nevermind}_e$,$\textit{album}_t$), ($\textit{Nevermind}_e$, $\textit{release}_t$), or ($\textit{Nevermind}_e$, $\textit{compact\_disc}_t$), where $e, t \in E, T$. As for the (exclusive) train-test split, we used at most 2k training pairs for each music type (although in the case of `song` these were 687 due to lack of enough song entities in the corpus), and evaluated on 500 entities, although again, the test size for the `song` type was smaller (229).

Then, under the intuition that there exists a linear function that *approaches* an unseen entity in our music model E to its most likely music type in the Google News corpus T, i.e., $\lambda(E) \approx T$, we train a linear regression model such that it minimizes

$$\min_{\lambda} \sum_{i=1}^{|E|} \|\lambda(\boldsymbol{e}_i) - \boldsymbol{t}_i\|^2 \tag{1}$$

We train four regression models, one for each of the music types considered. Then, evaluation consists of, given an input entity’s *text string*, applying each of the four models and assess which of them *approaches* the associated entity’s type vector the closest, and then assessing correctness. Then, for each test sample, the result is the ranked position among four possible candidates in which the

[9] `https://code.google.com/archive/p/word2vec/`

[10] These were collected manually by inspecting nearest neighbours to the different types considered in the Google News model.

correct type was placed. For example, for the input string $s =$'let it be'[11] (type `song`), we rank the closest vectors in T of $\lambda(s)$ by cosine distance, and set the position of the correct type to 1. If in this specific case, the `song` function yields the *second* most similar vector to 'let it be', the result is $[0, 1, 0, 0]$.

We evaluate the result in terms of Mean Reciprocal Rank (see [37] for a description of this and other Information Retrieval metrics in a Natural Language Processing setting), a metric which takes into account the position of the first valid candidate in a ranked list of options. Formally,

$$\mathrm{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}$$

where Q is a sample of experiment runs and $rank_i$ refers to the rank position of the *first* relevant outcome for the ith run.

Our results, provided in Table 5, computed over a sample of 100 entities per type, suggest that this is a promising approach, especially compared with approaches for similar tasks (hypernym discovery), which used much more training data coming from a wide range of resources such as Wikidata and the web [38]. Particularly encouraging are the results in the `record labels` entity type, although the fact that most record labels have words like 'label' or 'records' (e.g. *Epitaph Records*) most likely is being helpful to the model.

`artist`	`album`	`song`	`r.label`
0.59	0.52	0.54	0.64

Table 5. Mean Reciprocal Rank for the entity typing task

Visualizing *typed* vectors The ELMDist embeddings are trained over a music-specific corpus, and clustered by type. While theoretically the type similarity should already provide some kind of semantic compactness or community to embeddings of the same type, this might not be necessarily true due to the different contexts in which each musical entity may occur. In a purely qualitative way we intend to explore whether the *typing function* described above affects equally all musical types. In the visualization provided in Figure 1, we plot in red a sample of the original ELMDist vectors of the same type, and in green the same vectors (with labels for some of them for illustrative purposes) after combining them with the regression model. It can be clearly seen that artist and album vectors are scattered over the original space, which indicates that there is little corpus-based evidence for all artist names, for example, to be represented similarly. However, after training our typing function, artist names tend to cluster together, and while the semantics of these clusters remains to

[11] For multiword entities, we average the corresponding vectors of each token.

be explored in future work, it points towards an interesting "entity clustering" problem that can be approached similarly as in our proposed method. This phenomenon also applies to album names, although in this case there is clearly only one community. On the opposite side, songs and record labels seem to not benefit as much from the typing procedure, which may be due to the fact that their shared contexts are much more similar than in the other two entities. Another reason for the lackluster clustering in these two types is the much lower number of training pairs (due simply to ELMD2 having less entities of these types).

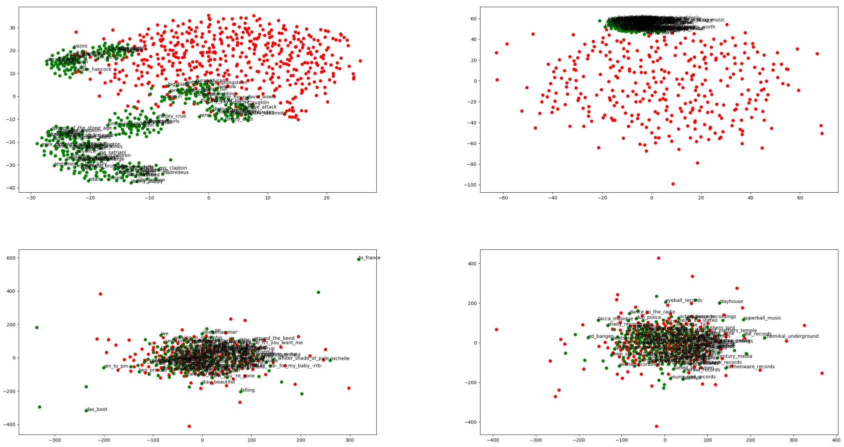

Fig. 1. Artist (top left), album (top right), song (bottom left) and record label (bottom right) embeddings before typing (red) and afterwards (green).

4 Conclusion and Future Work

In this paper, we have described and evaluated a novel vector space model at the *sense* level in the music domain. It comes from running the word2vec algorithm over an automatically disambiguated collection of music texts collected from last.fm, where music entities are automatically annotated leveraging the degree of agreement between three well-known Entity Linking and Word Sense Disambiguation systems. The model is evaluated qualitatively, in terms of artist similarity, and quantitatively, in terms of its usefulness for musical entity typing.

We believe our results show a promising avenue of work, improving Music Information Retrieval with textual information. For future work, we would like to incorporate larger corpora, probably coming from heterogeneous resources, and exploit current neural architectures both for entity disambiguation and for typing. In addition, a mixed model that combines musical information (e.g. in the form of audio descriptors) as well as semantic information coming from text corpora, seems to be a promising and unexplored direction. Finally, it would be interesting to learn different embeddings for each musical entity type and evaluate these entity-specific models as compared with models containing all entities.

5 Acknowledgements

We would like to thank the anonymous reviewers for their very helpful comments and suggestions for improving the quality of the manuscript. We also acknowledge support from the Spanish Minmistry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502) and under the TUNER project (TIN2015-65308-C5-5-R, MINECO/FEDER, UE).

References

1. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proceedings of the 26th International Conference on Neural Information Processing Systems, NIPS'13, Curran Associates Inc., USA, 2013, pp. 3111–3119.
2. J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1532–1543.
3. M. T. Pilehvar, N. Collier, De-conflated semantic representations, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 1680–1690.
4. A. Neelakantan, J. Shankar, A. Passos, A. McCallum, Efficient non-parametric estimation of multiple embeddings per word in vector space, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1059–1069.
5. F. Tian, H. Dai, J. Bian, B. Gao, R. Zhang, E. Chen, T.-Y. Liu, A probabilistic model for learning multi-prototype word embeddings, in: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin City University and Association for Computational Linguistics, Dublin, Ireland, 2014, pp. 151–160.
6. Y. Liu, Z. Liu, T.-S. Chua, M. Sun, Topical word embeddings, in: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI'15, AAAI Press, 2015, pp. 2418–2424.
7. C. Fellbaum, WordNet, Wiley Online Library, 1998.
8. R. Navigli, S. P. Ponzetto, Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network, Artif. Intell. 193 (2012) 217–250.
9. S. K. Jauhar, C. Dyer, E. Hovy, Ontologically grounded multi-sense representation learning for semantic vector space models, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 683–693.

10. M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. Hovy, N. A. Smith, Retrofitting word vectors to semantic lexicons, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 1606–1615.
11. J. Camacho-Collados, M. T. Pilehvar, R. Navigli, NASARI: a Novel Approach to a Semantically-Aware Representation of Items, in: Proceedings of NAACL, 2015, pp. 567–577.
12. A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 26, Curran Associates, Inc., 2013, pp. 2787–2795.
13. M. T. Pilehvar, R. Navigli, From senses to texts: An all-in-one graph-based approach for measuring semantic similarity, Artificial Intelligence 228 (2015) 95–128.
14. O. Etzioni, K. Reiter, S. Soderland, M. Sammer, T. Center, Lexical translation with application to image search on the web, Machine Translation Summit XI.
15. A. Fader, S. Soderland, O. Etzioni, Identifying relations for open information extraction, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11, Association for Computational Linguistics, Stroudsburg, PA, USA, 2011, pp. 1535–1545.
16. A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka, Jr., T. M. Mitchell, Toward an architecture for never-ending language learning, in: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI'10, AAAI Press, 2010, pp. 1306–1313.
17. C. Delli Bovi, L. Espinosa Anke, R. Navigli, Knowledge base unification via sense embeddings and disambiguation, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 726–736.
18. A. Swartz, Musicbrainz: A semantic web service, IEEE Intelligent Systems 17 (1) (2002) 76–77.
19. S. Oramas, M. Sordo, L. Espinosa-Anke, X. Serra, A Semantic-based Approach for Artist Similarity, in: Proceedings of the International Society for Music Information Retrieval Conference, Málaga, Spain, 2015, pp. 100–106.
20. M. Sordo, S. Oramas, L. Espinosa-Anke, Extracting Relations from Unstructured Text Sources for Music Recommendation, in: Proceedings of Natural Language Processing and Information Systems (NLDB), 2015, pp. 369–382.
21. S. Oramas, L. Espinosa-Anke, M. Sordo, H. Saggion, X. Serra, Information extraction for knowledge base construction in the music domain, Data & Knowledge Engineering 106 (2016) 70–83.
22. D. Gruhl, M. Nagarajan, J. Pieper, C. Robson, A. Sheth, Context and Domain Knowledge Enhanced Entity Spotting In Informal Text, in: The Semantic Web-ISWC 2009, Springer, 2009, pp. 260–276.
23. X. Zhang, Z. Liu, H. Qiu, Y. Fu, A Hybrid Approach for Chinese Named Entity Recognition in Music Domain, 2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing (2009) 677–681.
24. S. Oramas, L. Espinosa-Anke, M. Sordo, H. Saggion, X. Serra, ELMD: An automatically generated entity linking gold standard dataset in the music domain, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), 2016.

25. I. Iacobacci, M. T. Pilehvar, R. Navigli, Sensembed: Learning sense embeddings for word and relational similarity, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Beijing, China, 2015, pp. 95–105.
26. M. Manicini, J. Camacho-Collados, I. Iacobacci, R. Navigli, Embedding words and senses together via joint knowledge-enhanced training, arXiv prepring arXiv:1612.02703.
27. T. Mikolov, W.-T. Yih, G. Zweig, Linguistic regularities in continuous space word representations., in: HLT-NAACL, 2013, pp. 746–751.
28. D. P. Ellis, B. Whitman, A. Berenzweig, S. Lawrence, The quest for ground truth in musical artist similarity., in: ISMIR, Paris, France, 2002, pp. 170–177.
29. W. W. Cohen, W. Fan, Web-collaborative filtering: Recommending music by crawling the web, Computer Networks 33 (1) (2000) 685–698.
30. M. Schedl, P. Knees, G. Widmer, A web-based approach to assessing artist similarity using co-occurrences, in: Proceedings of the Fourth International Workshop on Content-Based Multimedia Indexing (CBMI'05), 2005.
31. B. Whitman, S. Lawrence, Inferring descriptions and similarity for music from community metadata, in: ICMC'02, 2002.
32. V. Shwartz, Y. Goldberg, I. Dagan, Improving hypernymy detection with an integrated path-based and distributional method (2016) 2389–2398.
33. T. Mikolov, Q. V. Le, I. Sutskever, Exploiting similarities among languages for machine translation, arXiv preprint arXiv:1309.4168.
34. R. Fu, J. Guo, B. Qin, W. Che, H. Wang, T. Liu, Learning semantic hierarchies via word embeddings, in: Proceedings of ACL, Vol. 1, Association for Computational Linguistics, 2014, pp. 1199–1209.
35. L. Tan, H. Zhang, C. Clarke, M. Smucker, Lexical comparison between wikipedia and twitter corpora by using word embeddings, in: Proceedings of ACL (2), Beijing, China, 2015, pp. 657–661.
36. S. Rodrıguez-Fernández, L. Espinosa-Anke, R. Carlini, L. Wanner, Semantics-driven recognition of collocations using word embeddings, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL): Short Papers, 2016, pp. 499–505.
37. J. Bian, Y. Liu, E. Agichtein, H. Zha, Finding the right facts in the crowd: factoid question answering over social media, in: Proceedings of the 17th international conference on World Wide Web, ACM, 2008, pp. 467–476.
38. L. Espinosa Anke, J. Camacho-Collados, C. Delli Bovi, H. Saggion, Supervised distributional hypernym discovery via domain adaptation, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 424–435.