# 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 an 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 with an estimated precision of above 0.9, 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 (in WordNet lingo, the entity's synset and sense). 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 paradigm [14,15,16], the truth is that current OIE systems are still too noisy and error-prone, and even approaches that have attempted an integration of 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.

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, predict the probability of it being any of three of the most common entities in the music domain, namely `artist`, `album` 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 a an automatic and fully disambiguated corpus in the music domain (Section 2.1). Second, we describe the linear transformation approach followed for assigning a music-specific type to any vector (Section 2.2). Finally, we provide evaluation results in Section 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]. 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.

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

Notable examples attempting Entity Linking (EL) (the task to assign 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 [21] or applying Hidden Markov Models for discovering musical entities in Chinese corpora [22].

In this work, we use as training data an extended version of the ELMD corpus [23], which stems from collection of biographies acquired from Last.fm[4]. The main idea behind the construction of ELMD is to take advantage of the fact that, while performance of generic EL systems may be lackluster due to their inability to account for linguistic idiosyncrasies in the music domain (abnormally long song or album names, existence of several valid abbreviations or acronyms for one single entity, and so forth), agreement in the prediction among an arbitrary number of systems can be used for accurate annotations. The original release of the ELMD corpus includes prediction outcomes from three systems, namely DBpedia Spotlight [24], Tagme [25] and Babelfy [26], from which an agreement score is given to each annotation, and then the predictions are homogeneously collapsed into one single *json* object (Figure 1 illustrates the main workflow of the disambiguation of ELMD).

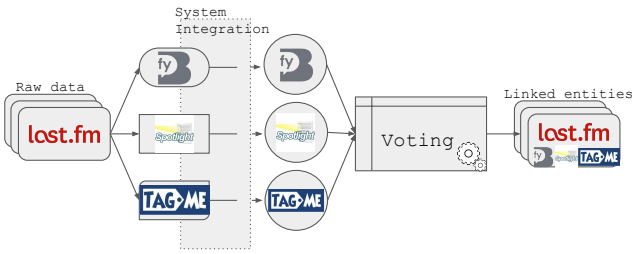


**Fig. 1.** Workflow for annotating the ELMD corpus

### 2.2 Training a sense-level embeddings model

Taking advantage of the mapping existing in ELMD between DBpedia `uris` and MusicBrainz ids (`mbids`), we follow [27,28] and, for each entity mention in the ELMD corpus, we concatenate its mention with its corresponding `mbid`, so that this "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.

[4] `http://last.fm`

the resulting disambiguated sentence would be[5]

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

The strategy to rely on agreement among EL systems used as black boxes, also simplifies the process, as in this experiment we take for granted that if an annotation exists, it will be correct. For instance, in [27], those annotations where their reference EL and Word Sense Disambiguation system resorted to the "most common sense" for annotation were discarded to ensure higher precision. It is important to recall, however, that in the original publication, the estimated precision in ELMD for musical entity annotations is reported to be around 0.94, which means that a certain degree of noise is inevitably introduced in the model.

We use the *gensim*[6] implementation of *word2vec* [29], and train a CBOW model of 300 dimensional vectors, filtering out tokens with frequency less than 3, with a context window of 5 tokens, and hierarchical softmax (usually a better performing algorithm for infrequent words).

## 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 to a better user experience. While artist similarity has been approached looking at score, acoustic or even cultural features [30], text-based approaches have also played an important role in this task. For instance, by computing co-occurrences of artist names [31], leveraging search engines result counts [32] or introducing further linguistic analysis in the form of ngram, part of speech and tf*idf information [33].

In this experiment, we asked 2 human judges to assess whether, given an input artist, the disambiguated nearest neighbours in the vector space were *similar*[7]. We randomly sampled 10 instances of `artists`, `albums`, `songs` and `record labels`. 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

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

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

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

that the judges looked at whether these record labels had 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 1, 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 2, where it can be clearly seen the difference between the quality of `artist` and `record label` vectors, 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 1.** 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 2.** 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 `track` 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 of album vectors were song vectors, and since we asked our judges to only consider for similarity the same entities (i.e. for albums, only albums), results for this entity type are not reported.

### 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?" [34]. 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 [35], 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 [35] 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 [36], Twitter language normalization [37], or for collocation discovery [38].

We follow this line of research, and construct an *entity matrix* $\mathbf{E} = [\boldsymbol{x}_1 \dots \boldsymbol{x}_n]$ and a *music type matrix* $\mathbf{T} = [\boldsymbol{y}_1 \dots \boldsymbol{y}_n]$, where $\mathbf{E}$ is our newly trained model, and $\mathbf{T}$ is the pretrained word2vec vectors on the Google News corpus[8]. 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. For instance, if we found the album *Nevermind* (by Nirvana), we would train with pairs such as ($Nevermind_e$,$album_t$), ($Nevermind_e$, $release_t$), or ($Nevermind_e$, $compact\_disc_t$), where $e, t \in E, T$. As for the (exclusive) train-test split, we used at most 2k training pair 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

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

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*, apply each of the four models and assess which of them *approaches* the associated entity's type vector the closest, and then assess correctness. Then, for each test sample, the result is the ranked position among four possible candidates in which the correct type was placed. For example, for the input string $s$ ='let it be'[9] (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, an Information Retrieval-derived metric which takes into account the position of the first valid candidate in a ranked list of options. Formally,

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

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

Our results, provided in Table 3, 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 [39]. 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 3.** Mean Reciprocal Rank for the entity typing task

## 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

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

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 Ministry 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: Advances in Neural Information Processing Systems, 2013, pp. 3111–3119.
2. J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014) 12.
3. M. T. Pilehvar, N. Collier, De-conflated semantic representations, EMNLP.
4. A. Neelakantan, J. Shankar, A. Passos, A. McCallum, Efficient non-parametric estimation of multiple embeddings per word in vector space, in: Proceedings of EMNLP, 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: COLING, 2014, pp. 151–160.
6. Y. Liu, Z. Liu, T.-S. Chua, M. Sun, Topical word embeddings., in: AAAI, 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, Artificial Intelligence 193 (2012) 217–250.
9. S. K. Jauhar, C. Dyer, E. H. Hovy, Ontologically grounded multi-sense representation learning for semantic vector space models., in: HLT-NAACL, 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 NAACL, 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: Advances in NIPS, Vol. 26, 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 EMNLP, 2011, pp. 1535–1545.
16. A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. H. Jr., T. M. Mitchell, Toward an Architecture for Never-Ending Language Learning, in: Proceedings of AAAI, 2010, pp. 1306–1313.
17. C. Delli Bovi, L. Espinosa Anke, R. Navigli, Knowledge base unification via sense embeddings and disambiguation, in: Proceedings of EMNLP, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 726–736.
URL http://aclweb.org/anthology/D15-1084
18. A. Swartz, Musicbrainz: A semantic web service, Intelligent Systems, IEEE 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.
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. 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.
22. 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) 677–681doi:10.1109/DASC.2009.27.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5380624
23. 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: Proccedings of the Language Resources and Evaluation Conference (LREC), 2016.
24. P. N. Mendes, M. Jakob, A. García-silva, C. Bizer, DBpedia Spotlight: Shedding Light on the Web of Documents, in: Proceedings of the 7th International Conference on Semantic Systems, 2011.
25. P. Ferragina, U. Scaiella, Tagme: on-the-fly annotation of short text fragments (by wikipedia entities), in: Proceedings of the 19th ACM international conference on Information and knowledge management, ACM, 2010, pp. 1625–1628.
26. A. Moro, A. Raganato, R. Navigli, Entity linking meets word sense disambiguation: a unified approach, Transactions of the Association for Computational Linguistics 2 (2014) 231–244.

27. I. Iacobacci, M. T. Pilehvar, R. Navigli, SensEmbed: Learning sense embeddings for word and relational similarity, in: Proceedings of ACL, Beijing, China, 2015, pp. 95–105.
28. M. Manicini, J. Camacho-Collados, I. Iacobacci, R. Navigli, Embedding words and senses together via joint knowledge-enhanced training, arXiv prepring arXiv:1612.02703.
29. T. Mikolov, W.-t. Yih, G. Zweig, Linguistic regularities in continuous space word representations., in: HLT-NAACL, 2013, pp. 746–751.
30. D. P. Ellis, B. Whitman, A. Berenzweig, S. Lawrence, The quest for ground truth in musical artist similarity., in: ISMIR, Paris, France, 2002.
31. W. W. Cohen, W. Fan, Web-collaborative filtering: Recommending music by crawling the web, Computer Networks 33 (1) (2000) 685–698.
32. 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 (CBMI05), 2005.
33. B. Whitman, S. Lawrence, Inferring descriptions and similarity for music from community metadata., in: ICMC, 2002.
34. V. Shwartz, Y. Goldberg, I. Dagan, Improving hypernymy detection with an integrated path-based and distributional method, ACL.
35. T. Mikolov, Q. V. Le, I. Sutskever, Exploiting similarities among languages for machine translation, arXiv preprint arXiv:1309.4168.
36. R. Fu, J. Guo, B. Qin, W. Che, H. Wang, T. Liu, Learning semantic hierarchies via word embeddings, in: Proceedings of ACL, Vol. 1, 2014.
37. 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.
URL `http://www.aclweb.org/anthology/P15-2108`
38. 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.
39. L. Espinosa-Anke, J. Camacho-Collados, C. D. Bovi, H. Saggion, Supervised distributional hypernym discovery via domain adaptation, in: Proceedings of EMNLP, 2016.