

Towards Automatic Ontology Alignment using BERT

Sophie Neutel^a, Maaïke H.T. de Boer^b

^aVrije Universiteit Amsterdam, De Boelelaan 1105, 1081 HV, Amsterdam, The Netherlands

^bTNO, Anna van Buerenplein 1, 2595 DA, The Hague, The Netherlands

Abstract

The job market is extremely flexible and constantly evolving. If information is represented in a machine-readable way, it is easier to add new terms or job titles and relate that to the existing terms. Several different representations of this field already exist, but those are not aligned yet. This paper examines the automatic alignment of two occupation ontologies - ESCO and O*NET - using Natural Language Processing methods. We specifically focus on a contextualized embedding model named BERT, and compare performance of five alignment systems. The novelty of this paper is twofold: 1) ontology alignment is applied in a real-world use-case in the labour market field; 2) BERT is applied for ontology alignment. It is found that, while their performance is not good enough yet to yield a useful alignment on their own, BERT-based embeddings mostly outperform word2vec-based embeddings. It is concluded that a hybrid approach is needed, where automatic alignment techniques are combined with manual alignment techniques, in order to improve coverage and eliminate errors.

Keywords

ontology, natural language processing, BERT, knowledge engineering

1. Introduction

The job market is extremely flexible and constantly evolving. In the Netherlands alone, new jobs are constantly arising, jobs are becoming obsolete, and existing jobs are changing. Add to this the fact that each country in the world has its own job market, as well as the fact that job markets are becoming increasingly international [1, 2], and it becomes clear that the information and knowledge contained in the job market is extensive and difficult to capture in its entirety. This poses a challenge to many processes that rely on this information and knowledge, such as recruitment, career guidance, and the development of curricula or policies.

Software tools are becoming increasingly important in this sector, in order to be able to monitor and manage the vast and ever-changing job market efficiently. Tooling could for example be helpful for analysts or policy makers to observe trends or it could help recruiters to find specific job profiles. A requirement for tools that exploit information regarding the job market is that this information needs to be represented in a machine-readable way. To represent the job market, all occupations and skills need to be mapped out, for example in databases

In A. Martin, K. Hinkelmann, H.-G. Fill, A. Gerber, D. Lenat, R. Stolle, F. van Harmelen (Eds.), *Proceedings of the AAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021)* - Stanford University, Palo Alto, California, USA, March 22-24, 2021.

✉ sophieneutel@gmail.com (S. Neutel); maaïke.deboer@tno.nl (M.H.T.d. Boer)

ORCID [0000-0002-2775-8351](https://orcid.org/0000-0002-2775-8351) (M.H.T.d. Boer)

© 2021 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).



CEUR Workshop Proceedings (CEUR-WS.org)

or ontologies. However, this is already a very complex task. Many organizations have created occupation and skill ontologies, but they all conceptualize and represent the domain of interest very differently. These differences make it difficult for different parties to exchange information. Ontology alignment can provide a solution for this. An alignment between two occupation ontologies can facilitate the exchange of information between different parties and the development of tools that exploit information regarding the job market.

This topic is investigated through the lens of a specific use case, namely the alignment of two existing, publicly available occupation ontologies: ESCO - European Skills, Competences, Qualifications and Occupations¹ - and O*NET - Occupational Information Network². An alignment is created for the purpose of developing a search tool that is able to match a given ESCO occupation with one or more O*NET occupations that encompass similar work activities and require similar skills.

This task is treated as a text similarity problem, rather than as a traditional ontology alignment problem: mappings between occupations are established based on the similarity score between occupations' descriptions. Specifically of interest is the question whether contextualized embedding models perform better than embedding models that do not take context into account. The novelty of this paper is twofold: 1) ontology alignment is applied in a real-world use-case in the labour market field; 2) the contextualized embedding model BERT is applied for ontology alignment.

The outline of this paper is as follows. Section 2 gives an overview of the existing literature on the topic of ontology alignment. Section 2.3 introduces the BERT model. Section 3 describes the data, the experimental setup and the evaluation method. The results are described in section 4 and discussed in section 5. Finally, section 6 concludes this thesis by describing how the results of this thesis can be used by the stakeholders, and by giving recommendations for future academic work.

2. Related Work

The term 'ontology' comes from the field of philosophy, where it describes the 'study of being'. In the fields of information science and artificial intelligence, the term 'ontology' is generally used to refer to a machine readable representation of a conceptualization of (a part of) the real world [3, 4, 5]. Ontologies are used to represent knowledge, often within a specific domain, in such a way that computers can reason over it and derive new information from established facts.

Section 2.1 gives a broad overview of the different types of approaches that could be taken to an ontology alignment problem. Section 2.2 describes several state-of-the-art ontology alignment systems, and introduces word embeddings as a useful tool for semantics-based ontology alignment.

¹<https://ec.europa.eu/esco>

²<https://www.onetcenter.org>

2.1. Ontology alignment approaches

Different approaches can be - and have been - taken towards the task of ontology alignment. Rahm and Bernstein [6] proposed a widely used classification of ontology alignment approaches. The main distinctions that can be made are the distinction between schema-level matching and instance-level matching, and the distinction between element-level matching and structure-level matching [6, 5, 7].

In schema-level matching, only schema information is used. Schema information is information at the concept level: schema-level matching is concerned with the concepts in an ontology, and not with the instances [6]. Schema information can include names, descriptions, relationship types, and structural information [6]. Instance-based matching makes use of instance-level information. Instance-level matching is especially useful in cases where there is limited schema information, or when there is no explicit schema information at all [6, 5]. Both schema-level matching approaches and instance-level matching approaches can be further subdivided into element-level matching approaches and structure-level matching approaches [6, 8, 5]. Element level matching approaches consider each entity in an ontology independent from the other entities in the ontology. Each single entity in the source ontology is matched to a single entity in the target ontology (where possible) [6, 5]. In structure level matching approaches, on the other hand, combinations of entities are matched to other combinations of entities [6, 5].

These approaches can then be further divided into specific types of approaches. Euzenat et al. [5] proposed a classification of different types of alignment approaches. Relevant to natural language processing are the string-based and language-based approaches. With string-based approaches the labels and descriptions (expressed in natural language) of elements in an ontology are matched by string similarity. Strings are viewed as sequences of characters, and the more similar the sequences of characters are to each other, the more likely they are to express the same concept [5]. In language-based approaches, natural language processing techniques are used to exploit (surface-level) properties of labels and descriptions in order to obtain a similarity score [5].

An important drawback of string-based approaches and a large number of language-based approaches is that their main focus is on surface-level similarity. They do not measure the underlying semantic similarity. In recent years, there has been a development towards language-based alignment approaches that focus on obtaining the underlying semantic similarity of concepts.

2.2. State-of-the-art ontology alignment methods

Harrow et al. [9] provide an overview of recent developments in ontology alignment for semantically enabled applications. One of the current challenges of ontology alignment is the ambiguity problem. Words can have different meanings depending on their context. Therefore, it is not sufficient to match concepts based on their surface-level linguistic features, such as class names or terms. In order to solve the ambiguity problem, context needs to be taken into account.

WordNet has commonly been used to determine the semantic similarity between elements [10]. However, over the last years, several new semantic similarity measures have been intro-

duced. Most notably, Zhang et al. [11] introduced word embeddings into the field of ontology alignment. In Zhang et al. [11], word2vec [12] embeddings are trained on Wikipedia, and are used to match entities based on the cosine similarity between entity names, entity labels, and entity comments. This method is evaluated on the OAEI 2013³ benchmark and conference track, as well as on three real-world ontologies. It was found that the matcher outperformed WordNet-based matchers in all test cases.

Dhouib et al. [13] align the Silex ontology⁴ - an ontology describing skills, occupations, and business sectors - with other ontologies in the same domain, one of which is ESCO. FastText [14] word embeddings are used to compute the similarity between concepts. A vector representation for each concept is obtained by averaging the word embedding vectors of all words in the concept's label. Cosine similarity is used to match each concept in the source ontology to the most similar concept in the target ontology. This system achieved state-of-the-art performance on an OAEI conference complex alignment benchmark [15].

Xue and Lu [16] propose a novel hybrid similarity measure for ontology alignment, which aggregates context-based, string-based, and dictionary-based similarity. Implemented with a Compact Brain Storm Optimization algorithm to reduce search space, they achieved a state-of-the-art performance.

Lu et al. [17] match concepts based on the semantic similarity of their labels. They combine cosine similarity with WordNet-based background knowledge. Their approach is evaluated on the OAEI 2016 benchmark⁵, and the performance is compared to the performance of the other systems that participated in OAEI 2016. It was found that their system ranks third in terms of precision, and ranks first in terms of recall and f-measure.

However, these state-of-the-art systems do not yet provide a fully satisfactory solution to the ambiguity problem. Words have different meanings depending on the context in which they occur. Word2vec based embeddings (which include fasttext embeddings) do not take context into account. Therefore, they are not able to differentiate between word senses nor can they capture fine-grained differences within a word sense. Take, for example, the occupation 'project manager': in the context of occupations and skills, a project manager at an IT company will have very different tasks and need very different skills from a project manager at a landscaping company. For an ontology alignment system to differentiate between two 'project managers', a word embedding model is needed that takes context into account.

2.3. BERT

BERT (abbreviation of Bidirectional Encoder Representation from Transformers) [18] is a transformer model [19] that has been trained to obtain deep bidirectional representations from unlabeled text. BERT provides contextualized embeddings, i.e. the same word gets different vectors depending on the context in which it occurs [18]. This implies that BERT could disambiguate between different word senses [20].

A transformer is a specific neural network architecture, which is typically used to handle sequential data - such as language. BERT's transformer architecture gives BERT an impor-

³<http://oaei.ontologymatching.org/2013>

⁴<https://www.silex-france.com/silex/>

⁵<http://oaei.ontologymatching.org/2016/>

tant advantage over other embedding models: deep bidirectionality. Other embedding models, such as ELMo - which has a bidirectional LSTM architecture - achieve bidirectionality by learning left-to-right context and right-to-left context separately, and then concatenating the two [21]. This is considered ‘shallow bidirectionality’ by Devlin et al. [18]: both left-to-right and right-to-left context are captured, but in such a way that the true context gets partially lost. In the transformer architecture, however, left-to-right and right-to-left context are captured simultaneously, thus capturing the complete context more accurately.

A significant disadvantage of BERT is the fact that it is not designed to provide representations for individual sentences. Many NLP tasks, including the ontology alignment task discussed in this paper, make use of sentence embeddings to represent the semantic content of a given text and to compute similarity between texts. At present, there is no clear-cut, widely accepted method to derive high-quality sentence embeddings from BERT [22, 23]. Common ways to derive fixed-length sentence embeddings from BERT are using the average of BERT’s output layer as the sentence representation, or using the [‘CLS’] token as a sentence representation [22, 24, 25, 23]. Reimers and Gurevych [22] evaluated both these approaches on seven semantic textual similarity (STS) tasks and on seven SentEval tasks. STS tasks measure the semantic similarity between two texts. SentEval [26] tasks are used to evaluate the quality of sentence embeddings. It was found that both the sentence representation that uses the [‘CLS’] token and the sentence representation that averages BERT embeddings yield poor results on the STS tasks.

In response to these issues, Reimers and Gurevych [22] introduced Sentence BERT (SBERT) an adaptation of pre-trained BERT that allows for semantically meaningful sentence embeddings, that can be compared using cosine similarity. In SBERT, a pooling layer is added to a pre-trained BERT network in order to obtain a fixed-size sentence embedding. SBERT is fine-tuned using siamese and triplet networks to update the network’s weights in such a way as to obtain semantically meaningful sentence embeddings. In the evaluation, SBERT outperformed other sentence embedding methods - including GloVe embeddings [27] and out-of-the-box BERT embeddings - on all seven STS tasks and on five out of seven SentEval tasks.

3. Experimental Setup

3.1. Data

As data, the occupation classifications ESCO and O*NET are used. Several mappings between occupation ontologies exist (e.g ESCO-ISCO), but an ESCO-O*NET mapping is still missing. While both ESCO and O*NET describe the same domain, they are very different in terms of structure, terminology, and semantics. Some of the differences and similarities are shown in table 1.

Table 2 shows which layers are present in the ESCO and O*NET hierarchies how many items each layer contains. Each ESCO layer differs in size from each O*NET layer. This indicates that the occupations are structured differently, and that they are divided into groups with differing levels of specificity. This can immediately be seen in the Major Groups in ESCO and O*NET: ESCO distinguishes far fewer major groups, pointing to differences in scope and level of detail between the two data structures. Some major groups seem like a good

| | ESCO | O*NET |
|-------------------|---|----------------------------------|
| Ontology Language | SKOS [28] | none |
| Granularity | detailed, fine-grained | smaller, less specific |
| Relations | parent-child, sibling, properties, associations | parent-child, sibling |
| Organization | hierarchy | hierarchy |
| Language | multilingual | English |
| Labels | preferred and optional | only 1 label |
| Writing style | lowercase, singular, complete sentences | capitalize, plural, omit subject |

Table 1

Differences and similarities between how ESCO and O*NET are structured and how occupations are labeled and described.

| Level | Esco | | O*NET | |
|---------|----------------------|------|----------------------|-----|
| Level 1 | Major groups | 10 | Major groups | 23 |
| Level 2 | Sub-major groups | 42 | Minor groups | 98 |
| Level 3 | Minor groups | 125 | Broad occupations | 459 |
| Level 4 | Unit groups | 426 | Detailed occupations | 867 |
| Level 5 | Occupations | 1701 | Detailed O*NET-SOC | 149 |
| Level 6 | Narrower occupations | 2128 | - | |

Table 2

Distribution of occupations over ESCO and O*NET layers

one-to-one match - such as `Armed forces occupations` (ESCO) and `Military Specific Occupations` (O*NET) - while for other major groups there is no good match. For example, an occupations under `Business and Financial Operations Occupations` in O*NET could fall under `Managers, Professionals, or Clerical support workers` in ESCO. O*NET seems to firstly divide occupations by topic, and then by function. The topic or area of work is on the major group level (`Business and Financial Operations`) and the function is specified on a lower level (sort of manager or analyst). ESCO seems to do this mostly the other way around. The function is specified on the major group level (`Managers`) and the topic or area of work is on a lower level (`communication manager` or `financial manager`).

3.2. Methods

A schema-level and element-based matching approach is taken. Individual ESCO occupations are matched with individual O*NET occupations, but only on the most specific occupations - i.e. the occupations at the bottom of their local hierarchy. Also, the ontology structure is disregarded completely. The ontology layers are treated as bags-of-occupations; the alignment takes place between a bag of ESCO occupations and a bag of O*NET occupations. There are two data points per occupation: the occupation label and the occupation description.

The ESCO occupations are divided into a training set (80%) and a test set (20%). Stratified sampling is used to ensure that each area of the ontology is sufficiently represented in the training and test data. The ten ESCO major groups are used as strata.

A very simple matching algorithm is used: each ESCO occupation is compared to each O*NET occupation. For each ESCO-O*NET occupation pair, a similarity score is calculated.

This results in a matrix displaying all similarity scores between all ESCO and O*NET occupations. Different methods to calculate the similarity score are used, as explained below:

Fasttext labels The baseline alignment system matches occupations based on the semantic similarity between their labels. Fasttext word embeddings are used to represent each token in the label with a 300-dimension vector. The entire label is then represented by taking the mean of all token vectors. Thus, each label is represented by a 300-dimension vector. The cosine distance between two labels is taken as the similarity score between the two corresponding ESCO-O*NET occupations.

Fasttext descriptions Each sentence in each description is represented by a 300-dimension vector using fasttext embeddings. A sentence vector is obtained by taking the mean of all token vectors.

BERT ‘CLS’ descriptions Each sentence in each description is embedded using BERT. The embedding of the [‘CLS’] token is extracted and used to represent the entire sentence. This results in a 768-dimension vector for each sentence.

BERT mean token descriptions Each sentence in each description is embedded using BERT. The embeddings of each individual token in the sentence are extracted, and the mean of these embeddings is used to represent the entire sentence. Thus, each sentence is represented by a 768-dimension vector.

SBERT descriptions This system represents each sentence in each occupation description using Sentence BERT (SBERT). As described in section 3, SBERT uses a pooling layer to create fixed-length sentence vectors which can be compared using cosine similarity.

3.3. Evaluation

There is no gold standard alignment to evaluate the systems’ output against. Therefore, the traditional evaluation metrics precision, recall, and f1-score cannot be used. Instead, the quality of the results is evaluated in terms of mean reciprocal rank (MRR). MRR indicates whether the matches found by the alignment systems are correct. A drawback of MRR is that it does not indicate whether all correct matches are found. To mitigate this issue, coverage is used as a secondary evaluation metric, to indicate the percentage of ESCO occupations for which at least one match was found.

The output of all systems is pooled, and annotated manually. For each ESCO-O*NET pair of occupations, a human judgement is made to determine whether this is indeed a correct match, or whether a system wrongly identified this pair as a match. There are four scenarios in which an ESCO occupation and an O*NET occupation are considered to be a match: 1) the occupations are exactly the same (*exact match*), 2) the occupations are very similar (*close match*), 3) the ESCO occupation is a subcategory of the O*NET occupation (*more specific match*), and 4) the ESCO occupation is a super-category of the O*NET occupation (*more general match*).

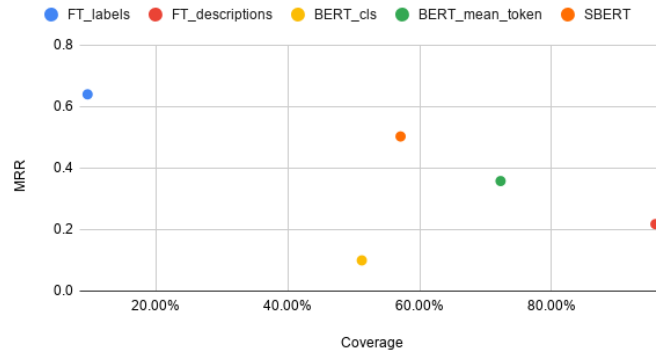


Figure 1: Each system’s mean reciprocal rank versus its coverage.

4. Results

The results of the experiments are shown in Figure 1. Label matching using cosine similarity between FastText (FT) embeddings scores the highest in terms of mean reciprocal rank, but has a very low coverage. The SBERT system achieves the second highest mean reciprocal rank, and has much higher coverage than the FastText label-matching system. Furthermore, the BERT CLS token system and the BERT mean token system have a lower mean reciprocal rank score than SBERT. However, the BERT mean token system does have a higher coverage. The system that uses FastText sentence embeddings has a high coverage, but performs poorly in terms of mean reciprocal rank. In the next subsection, an error analysis is described to get a better grasp of the results.

4.1. Error analysis

To gain further insight into the performance of each model, an error analysis is conducted on samples of each system’s output. An error occurs when a system matches two occupations that should not be matched. Three types of errors are distinguished:

- `Similar domain, different function (SimDDiff)`. Examples of this would be different functions in e.g. the domain of education, such as `sign language teacher (ESCO) → Teaching Assistants, Postsecondary (O*NET)`.
- `Different domain, similar function ((DifDSimF)`. Examples of this would be different types of technicians, such as `commissioning technician (ESCO) → Hydro-electric Plant Technicians (O*NET)`.
- `Different domain, different function (DifDDifs)`. Examples of this would be occupation pairs that are completely different from each other, such as `sailor (ESCO) → Floor Sanders and Finishers (O*NET)`.

Stratified samples are taken from each system’s erroneous matches, using the ESCO major groups as strata. All five error samples are annotated to indicate for each error - i.e. for each pair of occupations that should not have been matched - which of these three error types best

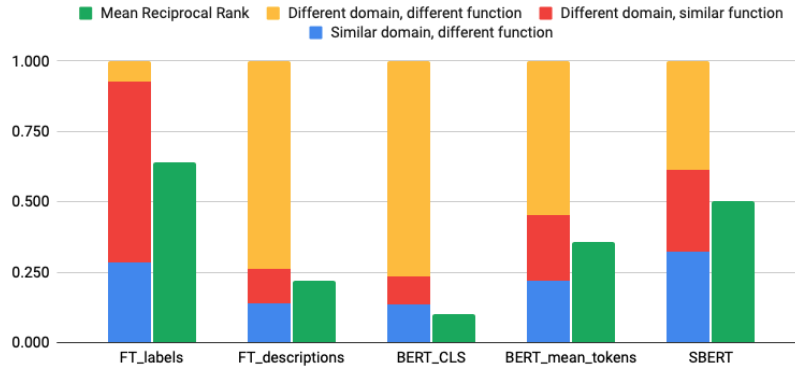


Figure 2: Proportion of error types and MRR per system.

describes it. The annotated error samples are then used to calculate the proportions of error types for each system. This is visualized in figure 2.

Looking only at figure 2, it seems that, for four out of five systems, it does not make a difference whether occupations are related by domain or by function. For all systems except the FT_labels system, these types of errors are represented fairly equally in the full sets of errors. Only the FT_labels system shows a clear tendency towards the $DifDSimF$ error type over the $SimDDifF$ error type. The main difference between the systems seems to be the proportion of errors where the occupations differ in both domain and function. An interesting observation is that there seems to be an inverse correlation between the proportion of unrelated errors and the mean reciprocal rank of each system. The systems with a higher mean reciprocal rank have a lower proportion of $DifDDifF$ errors, and a higher combined proportion of $SimDDifF$ errors and $DifDSimF$ errors.

In figure 2, the combined proportion of $SimDDifF$ errors and $DifDSimF$ errors is shown next to the mean reciprocal rank for each system. While the exact relation between the proportion of error types and mean reciprocal rank cannot be deduced from this, it is clear that the system with the highest mean reciprocal rank (FT_labels) also has the highest combined proportion of $SimDDifF$ errors and $DifDSimF$ errors - and therefore has the lowest proportion of $DifDDifF$ errors. When the five systems are ordered from highest to lowest mean reciprocal rank, this is the same order as if they were ordered from lowest to highest proportion of $DifDDifF$ errors.

5. Discussion

The results and the error analysis from the previous section suggest that the SBERT system yields the most promising results in the ontology alignment of the occupation ontologies ESCO and O*NET. The SBERT model used in this system has specifically been designed to yield high quality sentence embeddings. This is reflected in the fact that the SBERT system outperforms both the BERT_CLS system and the BERT_mean_token system. Furthermore, the errors made by the BERT mean token system and the SBERT system tend to be related by domain or by function more frequently than the errors made by the FT_description system. This suggests

that context sensitive embeddings - i.e. BERT-based embeddings - are better at estimating similarity and/or relatedness between descriptions than context independent embeddings - i.e. fasttext-based embeddings.

One would expect context sensitivity to allow systems to be able to distinguish between terms that are used in different senses. However, both the BERT_mean_token system and the SBERT system still erroneously match unrelated occupations that use similar or related terminology. In the current evaluation and error analysis method, it is unclear whether they do this less than the context independent FT_description matching system. Additional research would be required to quantify whether the BERT-based description matching systems are able to disambiguate words in different senses better than the fasttext-based description matching system. Following from this, it appears that the BERT_mean_tokens system and the SBERT system match related occupations, and not only similar occupations. The error analysis indicates that these systems cannot distinguish between similarity and relatedness.

Another interesting outcome of the matching experiments and the subsequent error analysis is that there appears to be a relation between error types and mean reciprocal rank. The proportion of completely unrelated errors seems to be an indication of a system's performance in terms of mean reciprocal rank. Further research should examine this observation further, to establish whether this is in fact a significant correlation and to determine what this means in relation to the matching systems.

For the purposes of this task, the SBERT system seems to be the most useful. It yields the second highest mean reciprocal rank, while also maintaining a reasonably high coverage. However, it is difficult to determine what the SBERT system's mean reciprocal rank of 0.503 means in practice. This is not a high score, meaning that the system makes a lot of errors and often does not rank a correct match in first place. With the future application in mind, none of these systems yield a good performance. A useful alignment has not been obtained using these methods. The recommended solution for this is to use a hybrid approach, which combines automatic and manual alignment. The SBERT system could be used to propose an initial mapping, which could then be corrected and extended manually. This would be less time-consuming than creating the entire alignment by hand.

It is difficult to compare these results to the state-of-the-art ontology alignment systems described in section 2, as the data set and evaluation method in this study are completely different from the data sets and evaluation methods used in those studies. A potential cause of the systems' poor performance could be found in the data. Both the ESCO dataset and the O*NET dataset are not very scientific in their structure. They have been designed in a very arbitrary way, and were not intended to be matched. The data is not very hierarchical and the classification of occupations is very different in the two data sets. As a result of this, structural information has been deemed to be unusable in this use case.

6. Conclusion and Future Work

In this paper an alignment between ESCO and O*NET is created using NLP techniques, in order to facilitate the exchange of information between different employment organizations and the development of tools that exploit information regarding the labour market. Similar occupations

were matched by embedding their descriptions and measuring the cosine similarity between them. Systems implementing context independent fasttext embeddings were compared with systems implementing context sensitive BERT embeddings in terms of their mean reciprocal rank and coverage. It was found that BERT's ['CLS'] token did not provide useful sentence embeddings. Fasttext sentence embeddings - obtained by taking the mean of the fasttext token embeddings of all tokens in the sentence - were found to establish the most matches, however the vast majority of these matches are incorrect. BERT sentence embeddings that were obtained by taking the mean of the BERT token embeddings of all tokens in the sentence found fewer matches, but also made fewer mistakes. Sentence BERT sentence embeddings were found to result in the best performance. While SBERT does not yield a ready-to-use alignment yet, it clearly outperforms the older approaches and provides a promising starting point for developing more effective alignment systems.

In future research, hybrid approaches will be explored, as well as the influence of the data set and the question of whether context sensitivity is actually beneficial for establishing similarity.

Acknowledgments

We would like to thank Piek Vossen for his supervision, the UWV for the data and the use case and the ERP Hybrid AI of TNO for their financial support in this use case.

References

- [1] C. Kuptsch, P. Martin, Actors and factors in the internationalization of labour markets, in: C. Kuptsch, D. Goux (Eds.), *The internationalization of labour markets*, International Institute for Labour Studies, 2010, pp. 115–134.
- [2] J. Cremers, M. Houwerzijl, *Internationalisering arbeidsmarkt/hrm-beleid* (2018).
- [3] B. Chandrasekaran, J. R. Josephson, V. R. Benjamins, What are ontologies, and why do we need them?, *IEEE Intelligent Systems and their applications* 14 (1999) 20–26.
- [4] B. Smith, C. Welty, *Ontology: Towards a new synthesis*, in: *Formal Ontology in Information Systems*, volume 10, ACM Press, 2001, pp. 3–9.
- [5] J. Euzenat, P. Shvaiko, et al., *Ontology matching*, volume 18, Springer, 2007.
- [6] E. Rahm, P. A. Bernstein, A survey of approaches to automatic schema matching, *the VLDB Journal* 10 (2001) 334–350.
- [7] E. Thiéblin, O. Haemmerlé, N. Hernandez, C. Trojahn, Survey on complex ontology matching, *Semantic Web* (2019) 1–39.
- [8] J. Kang, J. F. Naughton, On schema matching with opaque column names and data values, in: *Proceedings of the 2003 ACM SIGMOD Int. Conf. on Management of data*, 2003, pp. 205–216.
- [9] I. Harrow, R. Balakrishnan, E. Jimenez-Ruiz, S. Jupp, J. Lomax, J. Reed, M. Romacker, C. Senger, A. Splendiani, J. Wilson, et al., *Ontology mapping for semantically enabled applications*, *Drug discovery today* (2019).
- [10] F. Lin, K. Sandkuhl, A survey of exploiting wordnet in ontology matching, in: *IFIP Int. Conf. on AI in Theory and Practice*, Springer, 2008, pp. 341–350.

- [11] Y. Zhang, X. Wang, S. Lai, S. He, K. Liu, J. Zhao, X. Lv, Ontology matching with word embeddings, in: Chinese computational linguistics and natural language processing based on naturally annotated big data, Springer, 2014, pp. 34–45.
- [12] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781 (2013).
- [13] M. T. Dhoubib, C. F. Zucker, A. G. Tettamanzi, An ontology alignment approach combining word embedding and the radius measure, in: International Conference on Semantic Systems, Springer, Cham, 2019, pp. 191–197.
- [14] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146.
- [15] E. Thieblin, Task-oriented complex alignments on conference organisation, 2019. URL: https://figshare.com/articles/dataset/Complex_alignment_dataset_on_conference_organisation/4986368/8. doi:10.6084/m9.figshare.4986368.v8.
- [16] X. Xue, J. Lu, A compact brain storm algorithm for matching ontologies, IEEE Access 8 (2020) 43898–43907.
- [17] J. Lu, X. Xue, G. Lin, Y. Huang, A new ontology meta-matching technique with a hybrid semantic similarity measure, in: Advances in Intelligent Information Hiding and Multimedia Signal Processing, Springer, 2020, pp. 37–45.
- [18] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).
- [19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, 2017, pp. 5998–6008.
- [20] G. Wiedemann, S. Remus, A. Chawla, C. Biemann, Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings, arXiv preprint arXiv:1909.10430 (2019).
- [21] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, arXiv preprint arXiv:1802.05365 (2018).
- [22] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using siamese BERT-networks, arXiv preprint arXiv:1908.10084 (2019).
- [23] B. Wang, C.-C. J. Kuo, SBERT-WK: A sentence embedding method by dissecting BERT-based word models, arXiv preprint arXiv:2002.06652 (2020).
- [24] C. Sun, X. Qiu, Y. Xu, X. Huang, How to fine-tune bert for text classification?, in: China National Conference on Chinese Computational Linguistics, Springer, 2019, pp. 194–206.
- [25] J. Libovický, R. Rosa, A. Fraser, How language-neutral is multilingual bert?, arXiv preprint arXiv:1911.03310 (2019).
- [26] A. Conneau, D. Kiela, Senteval: An evaluation toolkit for universal sentence representations, arXiv preprint arXiv:1803.05449 (2018).
- [27] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proc. of the 2014 Conf. on EMNLP), 2014, pp. 1532–1543.
- [28] A. Isaac, E. Summers, Skos simple knowledge organization system, Primer, World Wide Web Consortium (W3C) 7 (2009).