A is the B of C:
(Semi)-Automatic Creation of Vossian Antonomasias

Johanna Rockstroh[1,8], Giada D'Ippolito[2,8], Nicolas Lazzari[3,8], Anouk M. Oudshoorn[4,8], Disha Purohit[5,8], Ensiyeh Raoufi[6,8] and Sebastian Rudolph[7,8]

[1]*University of Bremen*
[2]*University of Genova*
[3]*University of Bologna*
[4]*Technical University of Vienna*
[5]*Leibniz University Hannover*
[6]*University of Montpellier, LIRMM*
[7]*Technische Universität Dresden*
[8]*Team "Mordor" at the International Semantic Web Summer School 2023, Bertinoro*

Abstract

A *Vossian Antonomasia* (VA) is a stylistic device used to describe a person (or, more generally, an entity) in terms of a well-known person and a modifying context. For instance, the Norwegian chess world champion Magnus Carlsen was described as *"the Mozart of chess"* [1]. All VAs follow the pattern where a *source* (e.g., *"Mozart"*), is used to describe a *target*, (e.g., *"Magnus Carlsen"*), and the transfer of meaning is "channeled" through the use of the modifier *"of chess"*. Although this rhetorical figure is well-known, there has not yet been a dedicated study of targeted automatic or semi-automatic methods to generate and judge the appropriateness of VAs using large Knowledge Graphs (KGs) such as *Wikidata*. In our work, we propose the use of vector space embeddings – both KG-based and text-based – for producing VAs. For comparison, we contrast our findings with a purely LLM-based approach, wherein VAs are obtained from ChatGPT using a reasonably engineered prompt. We provide a publicly available GitHub repository[1] for the implementation of our method and a website[2] that allows testing the proposed methods.

1. Introduction

The question of whether computational methods can be used as creative devices can be traced back to the beginning of computers when Ada Lovelace wondered about the endless possibilities of automatic calculators [2]. Even though Artificial Intelligence techniques have largely been used in creative applications [3], the evaluation of such creative outputs remains problematic [4]. In this work, we propose the generation of Vossian Antonomasias (VAs) as a benchmark for exploring the creativity of AI methods.

---

[1]https://github.com/MordorISWS23/antonomasia

[2]The website is available at https://antonomasia.informatik.uni-bremen.de/. Note that for efficiency reasons, a restricted set of entities is available to users.

✉ rockstro@uni-bremen.de (J. Rockstroh); giadadippolito30@gmail.com (G. D'Ippolito); nicolas.lazzari3@unibo.it (N. Lazzari); anouk.oudshoorn@tuwien.ac.at (A. M. Oudshoorn); d.purohit@stud.uni-hannover.de (D. Purohit); ensiyeh.raoufi@lirmm.fr (E. Raoufi); sebastian.rudolph@tu-dresden.de (S. Rudolph)

VAs are a popular stylistic device for describing one entity by referring to another, typically in a witty and resourceful manner. A VA consists of three parts: a target entity A, a source entity B, and a modifier C, and is generally expressed as

A is the B of C.

A meaningful VA requires a non-trivial degree of creativity and extensive knowledge of the specifics of the target entity. One has to identify a set of salient characteristics of A that is similarly, or even more prominently realised by B. It is fundamental, however, that A and B differ when compared using the modifier C. For instance, in the sentence *"Nacho Figueras is the Brad Pitt of polo players"*, Ignacio Figueras, among the most famous polo players in the world, is compared to the actor Brad Pitt due to his appearance.[1]

VAs are often used in many journalistic genres and frequently appear in headings, as they can be both informative, enigmatic, and entertaining. In general, B is a well-known, widely recognized entity. Through its popularity, the writer encourages readers to classify A as similar to B, despite the difference C.

In this work, we present a method to automatically generate VAs by exploiting the latent semantic capabilities of vector space embeddings. We extract potential candidates for B by using SPARQL queries over Wikidata. We rely on a heuristic method to select entities that can be classified as popular. By relying on publicly available Knowledge Graph Embeddings (KGE) trained on Wikidata, we compute the vector representations for an arbitrary A, the identified set of B and a restricted set of C. We experiment with different operations between vectors to select the best B candidate.

In order to investigate the efficacy of each experiment, we compare the use of KGE with word embeddings obtained from large corpora of text [5]. Additionally, given the recent surge of Large Language Models (LLM) to mine creative analogies [6], we rely on ChatGPT [7] as an additional baseline. We evaluate each method through a user-evaluation study.

The paper can be summarised as:

1. Identification of a suitable pool of candidates that can serve as B elements;
2. Proposal of a novel method to automatically generate Vossian Antonomasias;
3. Evaluation of the proposed method using a user evaluation study;

The paper is organised as follows: in Section 2 we describe related work, which is followed by the presentation of the implemented method in Section 3. In Section 4, we discuss the results produced by the methods of Section 3 and present the outcomes of the user evaluation in Section 5. We finish by drawing conclusions and providing an outlook in Section 6.

---

[1]This and many more examples of VAs extracted from a newspaper corpus can be found at
https://vossanto.weltliteratur.net/emnlp-ijcnlp2019/vossantos.html

2. Related Work

There has been limited research on automatically detecting Vossian Antonomasias in written text. The authors of [1] demonstrated that by using Wikidata, they were able to overcome the shortcomings of available Named Entity Recognition (NER) tools and confirmed that VA is a linguistic and cultural phenomenon. Through quantitative VA explorations, they were able to capture the phenomenon as a whole, encompassing the source, target, and, when available, modifier. Their approach involves searching for a network of individuals interconnected by diverse modifiers, where the nodes can function as either sources or targets. This network aids in understanding hidden patterns of role models, revealing how they vary across countries and languages. However, a limitation they acknowledged is their reliance on the most prevalent pattern of VA, namely, *"the...of"* which resulted in the omission of numerous expressions extracted from the New York Times (NYT) corpus. For instance, notable phrases such as *"the American Oscar Wilde"* and *"Harlem's Mozart"* were overlooked despite their significance. Another approach for the extraction of VAs is presented by [8]. The focus is on the extraction of the target by using *coreference resolution* and visualising the connections between the source and target entities extracted in the VAs in form of a web demo. The authors of [9] use neural networks for the end-to-end detection of VAs resulting in two models: one for binary sentence classification and another for sequence tagging of all parts of the VA on the word level.

As opposed to the work described above, our approach focuses on the generation of VAs rather than their detection. Similarly to the approach of [1], our method only focuses on the pattern *"the...of"* by exploiting the latent semantic space of Knowledge Graph Embeddings and word embedding methods. Knowledge Graph Embedding compute a vectorial approximation of the originating Knowledge Graph through the use of various geometrical intuitions. In TransE [10] predicates and entities are modelled as translations in the vector space. Given a triple (h, r, t), the vector embedding of head entity h, predicate or relation r, and tail entity t are computed to minimise the quantity $|h + r - t|$ - i.e. t should be close to the $h + r$. Increasingly complex methods have been presented in literature [11].

Based on the distributional hypothesis, word embeddings [12] are used to compute vectors based on the distribution of words in large corpora of text, such as GloVe [13], where the representation is obtained using word co-occurrence statistics or word2Vec [14], where words with similar contextual distribution are approximated as similar vectors.

3. Methodology

This section details our proposed approach to automatically generate VAs, as outlined in Section 1. Figure 1 provides a high-level summary of our approach.

3.1. Wikidata as a Knowledge Resource

We rely on Wikidata to identify a set of candidates that can serve as *B* entities. This allows us to benefit from a large number of triples[2] with broad coverage of encyclopedic knowledge,

---

[2]According to Wikidata's statistics the knowledge graph currently contains 104,204,236 items. [15]
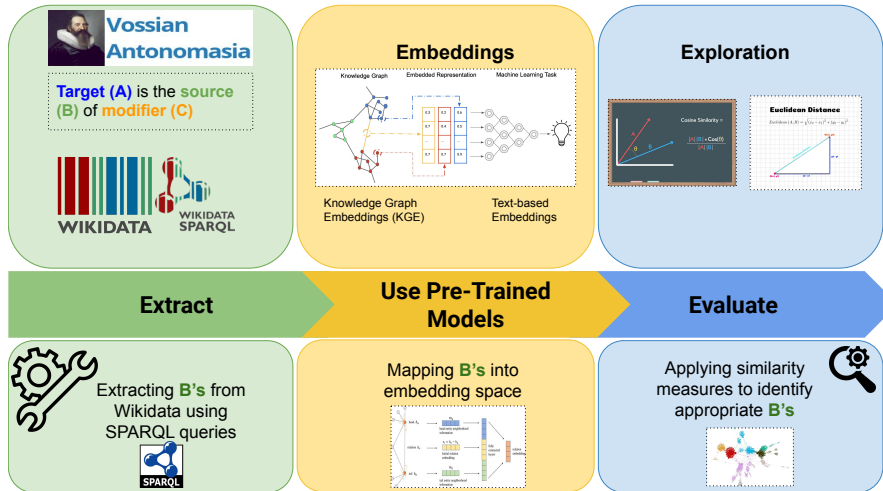
Figure 1: Vossian Antonomasia (VA) methodology: the approach we followed in our research work.

which enables us to select a sufficient sample of candidates for every component of the Vossian Antonomasias. Additionally, Wikidata provides a more structured and consistent data model when compared to similar resources, such as DBPedia[16]. Moreover, we can leverage the language-independent design of Wikidata as opposed to DBPedia [16] to ensure that the targets are widely popular.

We first extract the entities that will be used as *B* candidates by means of SPARQL queries.

We retrieve popular fictional characters and popular humans with the query of Listing 1. Given the large number of triples retrieved by both queries, they are executed on the Semantic Builders[3] SPARQL endpoint rather than on the regular Wikidata endpoint[4]. This allows us to overcome the querying timeout imposed by Wikidata and extract 3815 entities. The extracted entities are used as the set of candidates for *B* in a VA. We rely on a heuristic method to compute the popularity of an entity: the number of worldwide available Wikipedia articles in distinct languages for an entity as a proxy for its popularity. Given t the number of translations of one entity, we found $t \geq 70$ for real-world individuals and $t > 30$ for fictional characters to be a good estimate.

3.2. VA Generation using Vector Representations

We generate VA sentences by using geometrical transformations on the latent space provided by vector embeddings. Given an arbitrary *A*, we constrain the modifier *C* to be the occupation of the entity *A*. The underlying assumption of the proposed model is that, despite their different occupations, *A* and *B* need to be similar with respect to their salient features. For this reason, given a particular *A*, all those entities $b \in B$ that share the same modifier (i.e. the same occupation) are excluded from the pool of candidates. This brings us closer to ensuring the accurate selection of 'B' in accordance with the conditions specified in section 1.

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
PREFIX wd: <http://www.wikidata.org/entity/>
SELECT ?item ?itemLabel ?occupation ?sitelinks WHERE {
    ?item wdt:P31 <type>;
          wdt:P106 ?occupation;
          wikibase:sitelinks ?sitelinks .
    FILTER(<threshold> < ?sitelinks).
    SERVICE wikibase:label {bd:serviceParam wikibase:language
    ↪  "[AUTO_LANGUAGE],en".}}
```

Listing 1: SPARQL Query to extract candidate entities for *B*. *<type>* and *<threhsold>* are replaced by *wd:Q15632617* and 30 for fictional characters and *wd:Q5* and 70 for humans.

We propose two different methods for the selection of the best *B* candidate: a *translation-based* approach and a *projection-based* approach. The translation-based approach follows the intuition of *TransE* and *word2vec*: given $\vec{x}$, $\vec{y}$ the vector representations of two arbitrary entities and given $\vec{c}$ the vector that represents a predicate holding between $\vec{x}$ and $\vec{y}$, it has been observed that $\vec{x} + \vec{c} \approx \vec{y}$.

Given $\vec{a} = (a_1, \ldots, a_d)$ the embedding vector of an arbitrary target entity A, $\mathscr{B}$ the set of embedding vectors of each candidate entity for *B*, and $\vec{c} = (c_1, \ldots, c_d)$ the embedding vector of the predicate *C* that denotes the *occupation* of an entity (i.e. P106 in Wikidata), the translation-based method first disregards (i.e., "subtracts") A's occupation C obtaining:

$$\vec{a'} = \vec{a} - \vec{c} \tag{1}$$

Then, we define the fitness f (where smaller is better) of a candidate $\vec{b'} \in \mathscr{B}$ as the Euclidean distance between $\vec{a'}$ and $\vec{b'}$, i.e.,

$$f(\vec{a}, \vec{b'}) = |\vec{a'} - \vec{b'}| = \sqrt{\sum_{i=0}^{d} (a_i' - b_i')^2} \tag{2}$$

where d is the dimension of the vector embedding space.

The projection-based method relies on a different assumption. Informally, we would like to compute the fitness $f(\vec{a}, \vec{b'})$ on a subspace of the whole embedding space where every information related to the occupation of the entities is ignored. We compute the projection of $\vec{a}$ and $\vec{b'}$ to this subspace, which is a hyperplane perpendicular to $\vec{c}$, using

$$\pi_{\vec{c}}(\vec{x}) = \vec{x} - \left(\frac{\vec{x} \circ \vec{c}}{\vec{c} \circ \vec{c}} \right) \vec{c} \tag{3}$$

where $\circ$ denotes the inner product and $\vec{x}$ is either $\vec{a}$ or $\vec{b'}$. The fitness function f is hence adjusted to the cosine distance between the projections of $\vec{a}$ and $\vec{b'}$, formally

$$f(\vec{a}, \vec{b'}) = \frac{\pi_{\vec{c}}(\vec{a}) \circ \pi_{\vec{c}}(\vec{b'})}{|\pi_{\vec{c}}(\vec{a})||\pi_{\vec{c}}(\vec{b'})|} \tag{4}$$

(a) Equation 2 (translation-based) illustrated.

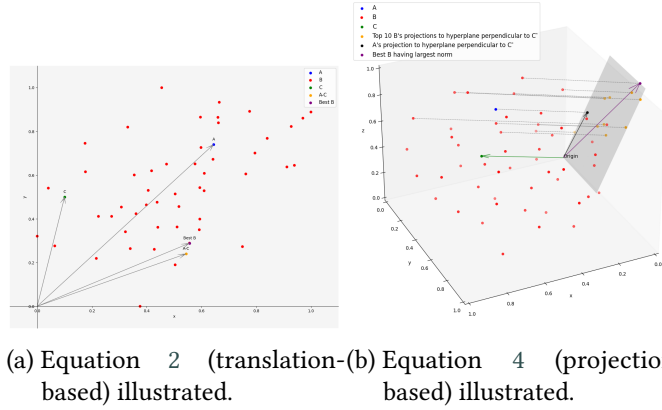(b) Equation 4 (projection-based) illustrated.

Figure 2: Illustration of the fitness function *f* for the translation-based method (Figure 2a) and the projection-based method (Figure 2b).

The fitness functions in Equation (2) and Equation (4) can be seen as similarity functions between two entities. A suitable *B* can be extracted by taking the entity that minimises such distance. Intuitively, in the translation-based approach, the vector representing *B* is supposed to be similar to the one of *A* after we "translate away" or "subtract" the characteristics pertaining to *C* using Equation (1), while in projection-based method by applying Equation (3), we "project away" such characteristics.

As addressed in Section 1, a good VA needs to be a creative sentence. While it is difficult to assess creativity in an objective manner, it has been argued that among the many characteristics, a creative output needs to display novelty when compared to others [17]. Given a set of entities $\hat{B}$ that minimises the fitness function *f*, we propose to further rank such entities by using their L_1 norm. The intuition is that among all the candidates in $\hat{B}$, the ones with a greater distance from the origin of the vector space are the most "extremal" ones.

Figure 2 depicts a simplified illustration of Equation 2 and Equation 4. Using t-SNE [18], we reduce the dimensionality of embedding vector $\vec{a}$ of a sample entity A, a set $\mathscr{B}$ of embedding vectors of each candidate entity for *B*, and $\vec{b} \in \hat{B}$ vectors.

Given a particular entity *A* and the corresponding entity *B*, selected with one of the proposed methods, we generate an assertional VA following template: `{A} {verb} the {B} of {C}`. If the entity *A* has an entry in Wikidata that certifies its death, we set verb to *was*, else we use *is*.

3.3. Purely LMM-based Baseline via ChatGPT

Finally, we use ChatGPT [7], a Large Language Model, as a baseline for VA generation. Through extensive experimentation, we found the prompt that obtains the best result to be

Following the discussion of Section 1, we argue that an effective prompt for VA needs to display the following properties:

- *B* should not share characteristics with *C*;
- *A* and *B* should share at least one salient characteristic;
- *A* and *B* should be popular enough to draw the analogy in the context of *C*

Provide 10 Vossian Antonomasias for *<Name of A>*, where she is equated with another person. Each of the phrases should have the structure "*<Name of A>* is the [person name] of [profession]", where [profession] must not characterize [person name]. Provide a very short justification for each example.

Table 1
ChatGPT prompt used to obtain a VA.

4. Experimental Setting

As briefly addressed in Section 1, we experiment with two different methods to obtain vector representations of an entity: Knowledge Graph Embeddings (KGE) and Word Embeddings (WE).

We employ TransE [10] as KGE method. We directly reuse the publicly available model shared by GraphVite [19] and trained on the Wikidata-5M dataset [20]. For the WE method, we employ word2vec [14] and GloVe [13] provided by gensim [5].

Finally, we leverage the use of meta-embedding techniques, i.e. combining different embedding methods together [21], to exploit the main advantages of both methods. We combine KGE and WE by means of concatenation and averaging. When averaging two vectors with different dimensionality, we apply zero-padding [21]. Note that, even though they are supposed to converge to a similar semantic, the latent space represented by a method might differ drastically from other methods. To prevent a drastically higher influence of one method over the other, we normalise both methods by their L_2 norm before combining them.

To allow interested readers to try out the different methods themselves, we set up a demonstrator website available at https://antonomasia.informatik.uni-bremen.de/.

5. Results and Evaluation

Due to the highly diverse nature of VAs, we decided to test the quality of the output with human evaluation. A small selection of examples generated by the methods described in Section 3 is presented in Table 2.

The selection on Table 2 shows that ChatGPT, while being very creative when it comes to the description of the domain, does not perform well in identifying a proper *B* that does not share characteristics with the modifier *C*, such as in the sentence "Bill Gates is the Einstein of Societal Transformation". This phenomenon particularly occurs with politicians or writers. Despite the explicit request in the prompt of Table 1, ChatGPT did not manage to adapt the chosen entity. The results generated by the KGE, WE and their combination mostly meet the mentioned criteria, even though some exceptions occur, such as in the sentence "Angela Merkel is the Eva Braun of politics.".

5.1. User Evaluation

The method described in Section 3 allows combining several different techniques to generate a VA. For the user evaluation, after manual experimentation, we restricted the set of techniques to
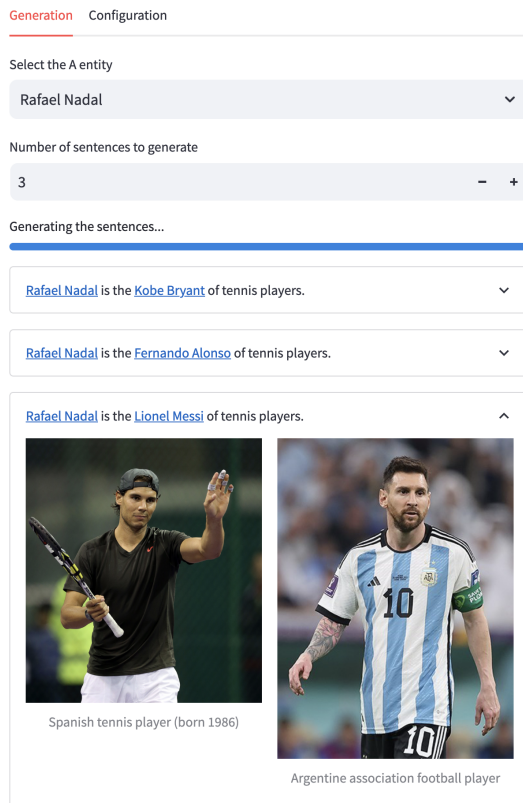
Figure 3: Screenshot of the demonstrator website

Table 2

Good and bad examples generated by three different methods. Each rating refer to an evaluation where the user has complete knowledge and familiarity with both entities.

	Sentence	Score
KGE	Nelson Mandela was the Cameron Diaz of politics.	5
	Mark Twain was the Darth Vader of journalists.	1
WE	Bill Gates is the Hugh Grant of entrepreneurs	5
	Angela Merkel is the Julian Assange of politics.	1
Meta	Mark Twain was the Hemingway of Satirical Prose.	5
	Angela Merkel is the Eva Braun of politics.	1
LLM	Mark Twain was the Neil Armstrong of journalists.	5
	Bill Gates is the Einstein of Societal Transformation.	1

the ones described in Table 3. The presented selection allows us to evaluate the importance of different assumptions, such as whether the methods based on the distributional hypothesis can complement content-based methods. Moreover, we are able to assess whether the presented methods can overcome the issues of ChatGPT, namely the difficulty of selecting *A* and *B* from a different domain dictated by *C*.

Table 3

Methods employed in the user evaluation study. The meta embedding obtained by concatenating TransE and word2vec is written as TransE ⊕ word2vec.

Method	Fitness function
ChatGPT	
TransE ⊕ word2vec	Project (Equation (4))
TransE ⊕ word2vec	Translate (Equation (2))
TransE	Translate (Equation (2))
TransE	Project (Equation (4))
word2vec	Translate (Equation (2))
word2vec	Project (Equation (4))

We identify six individuals that will be used as *A*: *Nelson Mandela, Angela Merkel, Mark Twain, Albert Einstein, Bill Gates* and *Ronald Reagan*. Those individuals are both part of the entities extracted using the query in Listing 1 and the real-world samples from the New York Times [1] and Der Umblätterer[5] corpora. As mentioned in Section 3, we contain *C* to the profession of the entity.

To recruit participants for our study, we distributed a flyer as advertisement and shared it with colleagues and friends who themselves distributed this further. The study can be done online without any supervision. Due to the specificity of Vossian Antonomasia, we present a definition on the front page of the study:

> Vossian antonomasias refer to someone by a special characteristic instead of their name.For example, calling Bill Gates "the Henry Ford of the computer age" high-lights his influence as entrepeneur and his effect on the development of technology. It is a way to describe someone by an important quality they possess.

The study was open for one week. We provide each participant with 21 sentences. We randomly sample 3 *A*s from the set described above and provide 7 VAs, one for every generation method of Table 3. The selected VAs are randomly sampled from the top 10 sentences identified by the method used. The participant is asked to judge each VA on three aspects: how well the description fits, how understandable it is and how original the VA is. These three aspects can be ranked on a Likert Scale ranging form 1 to 5. After that, the knowledge about the source and the target will be inquired in the form of questions: we ask how well the participant knows the individuals. The possible answers are *I know who that person is, I have heard of the name but I cannot relate it to anything*, and *I have never heard of that name before*. This ensures a distinction between a negative rating caused by ignorance of the output's components and the lack of a proper connection between the source, the target and the modifier.

5.2. Results

Through the user-evaluation test, we obtain a set of 207 human evaluations on automatically generated VAs, provided by 29 unique annotators. The sentences presented to the participants

---

[5]https://www.umblaetterer.de/datenzentrum/vossianische-antonomasien.html

Rate Vossian Antonomasias

Please rate the following sentences with **1** being the best and **5** being the worst evaluation.

Albert Einstein was the Galileo of Thought Experiment.

How well does this description fit?

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

How understandable is this description?

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

How original is this description?

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Figure 4: Frontend of the user study for rating the VAs.
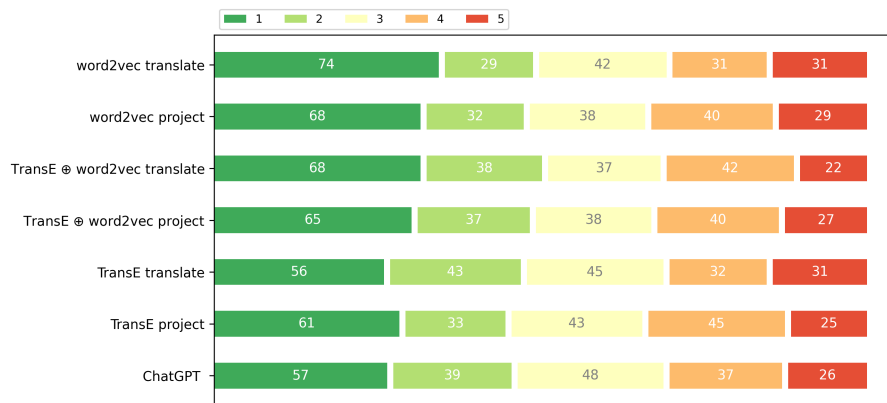

Figure 5: Distribution of the mean rating among the methods of Table 3 with *1* being the best and *5* being the worst rating.

are repeated to avoid a random evaluation. The inter-annotator agreement, computed using Cohen's Kappa score [22], is 0.0491 on average. Such a low score highlights the difficulty in evaluating VAs since they greatly depend on the reader's knowledge, cultural reference and degree of familiarity with the selected subject.

In Figure 5 the distribution of ratings among the methods reported in Table 5 is reported. Intuitively, whose distribution is skewed towards low ratings (represented in green) should be considered the best-performing method. Using the translation-based meta-embedding method outperforms all the other methods. Interestingly, the baseline provided by ChatGPT performs worse than any other method. While this might be reconducted to the lack of explicit VA-related knowledge that the underlying LLM is based on, it can also be argued that the prompt that we propose to use is not perfectly suited for this task. We will further address this aspect in Section 6. At first glance, the best method turns out to be word2vec using the translation technique, which results in 74 VAs rated with a score of 1. It needs to be argued, however, that using this

method results also in a high variance in the results. Indeed, the same method is also the one that obtains the highest amount of low-rated VAs.

Table 4
Average rating for each method. Lower is better. Best result is represented in bold.

	Method	Average rating
	ChatGPT	2.69
Translate Project	TransE ⊕ word2vec	2.64
	TransE	2.71
	word2vec	2.66
	TransE ⊕ word2vec	**2.57**
	TransE	2.70
	word2vec	2.59

When taking into account the whole distribution, the projection-based method TransE achieved the best performance: the lowest number of "bad" VAs is obtained with such a method. Indeed, this is the method that achieves the best overall performances, as can be seen in Table 4.

Table 5
Overview of the mean rating for each user question. The mean rating for each confidence rating (μ_1, μ_2, and μ_3) is reported alongside the overall mean rate $\mu_{1\cup2\cup3}$ for each question. The best results for each criterion are highlighted in bold. Lower values indicate better performance.

	Method	Fit				Understand				Original			
		μ_1	μ_2	μ_3	μ	μ_1	μ_2	μ_3	μ	μ_1	μ_2	μ_3	μ
	ChatGPT	2.21	2.09	2.81	2.51	2.42	2.27	3.22	2.80	2.63	3.27	2.73	2.77
Translate Project	TransE ⊕ word2vec	2.26	2.07	2.67	2.42	2.16	2.14	3.15	**2.64**	2.47	3.14	2.97	2.88
	TransE	1.93	2.38	2.67	2.45	2.14	2.00	3.41	2.84	2.50	2.46	3.15	2.84
	word2vec	1.88	2.08	2.63	**2.30**	2.00	2.62	3.34	2.83	2.50	2.77	3.08	2.86
	TransE ⊕ word2vec	2.17	2.60	2.33	2.35	2.22	2.87	2.89	2.71	2.39	2.67	2.81	2.67
	TransE	1.70	2.71	2.55	2.38	1.85	3.29	3.23	2.87	2.40	3.00	3.10	2.87
	word2vec	1.89	2.07	3.00	2.48	2.11	1.93	3.31	2.65	2.11	2.93	2.88	**2.65**

Table 5 provides an overview of the mean rating μ_i for each method, where i represents the set of sentences for which an evaluator expressed specific confidence in the knowledge of A and B. Those results are complementary to the ones of Figure 5. Interestingly, the best overall method, projection-based TransE, does not classify as the best in any specific user question. Depending on the target task, one model can be considered better than the other, even though projection-based TransE guarantees the most consistent results.

6. Conclusions and Discussion

We looked at generating Vossian Antononmasias by using embeddings and LLM as a way to characterize the creativity of AI. Our approach has resulted in creative examples of VAs, which proves that both methods are suitable for solving such tasks. Since the lack of a clear definition

of creativity prevents a quantitative evaluation of the results, we conducted a manual qualitative analysis of the results which highlighted several different weaknesses. The information bias that is inherent to Wikidata results in VAs that are mostly focused on Western culture. While this might be tampered by penalising some entities, we argue it would only partially solve the issue. A different approach, which takes into account the semantic representation of each entity, can help overcome such issues, making the creation process transparent and explainable.

The human evaluation described in Section 5 provides meaningful insights into the effectiveness of our methods. Firstly, they generally perform better than the ChatGPT baseline, which fails in the generation of original and understandable VAs. Moreover, the results of Figure 5 and Table 5 show how the use of Knowledge Graph Embeddings is generally to be preferred over the methods, such as word embeddings and meta-embeddings. However, the low inter-annotator agreement shows that rating Vossian Antonomasias is highly subjective and most probably depends on not only the knowledge of an entity but also knowledge of the domain the entity refers to. This could be addressed by filtering the human annotators into groups according to their domain knowledge before rating the sentences.

The focus on the occupation as a similarity measure resulted in several complications, such as the use of semantically similar occupations like *television actor* and *actor* in the generated sentences. Additionally, since some entities hold multiple occupations, a more accurate estimation of their primary occupation needs to be investigated. A possible solution is to aggregate the representation vectors of all their occupations instead of selecting a single occupation. Similarly, fictional characters are sometimes compared to their real-life actors. A possible solution is to impose a minimum distance between vectors that are too close. An orthogonal solution is to consider other criteria when comparing entities, such as achievement or awards. Apart from famous people, famous locations or events along with their appropriate modifiers could be added to increase the sample size and achieve a greater variability in the results.

The mentioned limitations of the evaluation and sampling of entities show that the generation of Vossian Antonomasia with an open-domain approach proves to be rather difficult. Instead, we suggest focusing on specific domains, thereby using fine-tuning of the embeddings or different embedding methods to overcome the mentioned shortcomings.

Additionally, we envision integrating an LLM in the pre-evaluation step by evaluating the VAs that have been generated by combining our proposed methods. The idea is to list the salient similarities and differences between *A* and *B* for a given VA by looking at their characteristic properties. Following recent LLM-prompting studies [23, 24], we plan on performing additional manual or automatic prompt engineering [25, 26]. This can lead to more effective results, since any change in the prompts may affect significantly the quality of the output.

Moreover, an interesting approach is to perform knowledge injection [27, 28] into a LLM, following a neuro-symbolic approach to use the language model's potential and control for the criteria defining Vossian Antonomasia. The knowledge available to an LLM like ChatGPT is currently limited regarding whether an entity has recently died or is a fictional character based on the time it was trained. The injection of structured knowledge can help overcoming this issue.

Acknowledgements

References

[1] F. Fischer, R. Jäschke, 'The Michael Jordan of Greatness'— Extracting Vossian Antonomasia from Two Decades of *The New York Times*, 1987–2007, Digital Scholarship in the Humanities (2019). URL: https://doi.org/10.1093/llc/fqy087. doi:10.1093/llc/fqy087.

[2] L. F. Menabrea, Sketch of the Analytical Engine invented by Charles Babbage, Esq., in: Ada's Legacy: Cultures of Computing from the Victorian to the Digital Age, 1843.

[3] N. Anantrasirichai, D. R. Bull, Artificial Intelligence in the Creative Industries: A Review, Artif. Intell. Rev. 55 (2022) 589–656. URL: https://doi.org/10.1007/s10462-021-10039-7. doi:10.1007/s10462-021-10039-7.

[4] R. Wingström, J. Hautala, R. Lundman, Redefining Creativity in the Era of AI? Perspectives of Computer Scientists and New Media Artists, Creativity Research Journal (2023) 1–17.

[5] R. Řehůřek, P. Sojka, Software Framework for Topic Modelling with Large Corpora, in: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, ELRA, Valletta, Malta, 2010, pp. 45–50. http://is.muni.cz/publication/884893/en.

[6] B. Bhavya, J. Xiong, C. Zhai, Cam: A Large Language Model-based Creative Analogy Mining Framework, in: Proceedings of the ACM Web Conference 2023, WWW '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 3903–3914. URL: https://doi.org/10.1145/3543507.3587431. doi:10.1145/3543507.3587431.

[7] OpenAI, ChatGPT, https://openai.com, 2021. Version GPT-3.5.

[8] M. Schwab, R. Jäschke, F. Fischer, "Who is the Madonna of Italian-American Literature?": Extracting and Analyzing Target Entities of Vossian Antonomasia, in: Proceedings of the 7th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, Association for Computational Linguistics, 2023, pp. 110–115. URL: https://sighum.files.wordpress.com/2023/03/latech-clfl-2023-unofficial-proceedings.pdf.

[9] M. Schwab, R. Jäschke, F. Fischer, "The Rodney Dangerfield of Stylistic Devices": End-to-End Detection and Extraction of Vossian Antonomasia Using Neural Networks, Frontiers in Artificial Intelligence 5 (2022). URL: https://doi.org/10.3389/frai.2022.868249. doi:10.3389/frai.2022.868249.

[10] A. Bordes, N. Usunier, A. García-Durán, J. Weston, O. Yakhnenko, Translating Embeddings for Modeling Multi-relational Data, in: C. J. C. Burges, L. Bottou, Z. Ghahramani, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada,

United States, 2013, pp. 2787–2795. URL: https://proceedings.neurips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html.

[11] S. Choudhary, T. Luthra, A. Mittal, R. Singh, A Survey of Knowledge Graph Embedding and Their Applications, CoRR abs/2107.07842 (2021). URL: https://arxiv.org/abs/2107.07842. arXiv:2107.07842.

[12] F. Almeida, G. Xexéo, Word embeddings: A survey, CoRR abs/1901.09069 (2019). URL: http://arxiv.org/abs/1901.09069. arXiv:1901.09069.

[13] J. Pennington, R. Socher, C. D. Manning, Glove: Global Vectors for Word Representation, in: A. Moschitti, B. Pang, W. Daelemans (Eds.), Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, ACL, 2014, pp. 1532–1543. URL: https://doi.org/10.3115/v1/d14-1162. doi:10.3115/v1/d14-1162.

[14] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, in: Y. Bengio, Y. LeCun (Eds.), 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings, 2013. URL: http://arxiv.org/abs/1301.3781.

[15] Wikidata statistics, https://www.wikidata.org/wiki/Wikidata:Statistics, 2023. Accessed on 2023-06-16.

[16] D. Abián, F. Guerra, J. Martínez-Romanos, R. T. Lado, Wikidata and DBpedia: A Comparative Study, in: International KEYSTONE Conference, 2017.

[17] G. Ritchie, Assessing Creativity, in: Proc. of AISB'01 Symposium, 2001.

[18] L. Van der Maaten, G. Hinton, Visualizing Data using t-SNE., Journal of Machine Learning Research 9 (2008).

[19] Z. Zhu, S. Xu, M. Qu, J. Tang, Graphvite: A High-Performance CPU-GPU Hybrid System for Node Embedding, in: The World Wide Web Conference, ACM, 2019, pp. 2494–2504.

[20] X. Wang, T. Gao, Z. Zhu, Z. Zhang, Z. Liu, J. Li, J. Tang, KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation, Trans. Assoc. Comput. Linguistics 9 (2021) 176–194. URL: https://doi.org/10.1162/tacl_a_00360. doi:10.1162/tacl_a_00360.

[21] D. Bollegala, J. O'Neill, A Survey on Word Meta-Embedding Learning, in: L. D. Raedt (Ed.), Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022, ijcai.org, 2022, pp. 5402–5409. URL: https://doi.org/10.24963/ijcai.2022/758. doi:10.24963/ijcai.2022/758.

[22] J. Cohen, A Coefficient of AgI see you're on the paper, I updated the table and will proofread everything now, do you think that are particulreement for Nominal Scales, Educational and Psychological Measurement 20 (1960) 37–46.

[23] X. Liu, Y. Zheng, Z. Du, M. Ding, Y. Qian, Z. Yang, J. Tang, GPT Understands, Too, arXiv preprint arXiv:2103.10385 (2021).

[24] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, D. C. Schmidt, A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT, arXiv preprint arXiv:2302.11382 (2023).

[25] L. Reynolds, K. McDonell, Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm, in: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems, 2021, pp. 1–7.

[26] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, J. Ba, Large language Models Are Human-Level Prompt Engineers, arXiv preprint arXiv:2211.01910 (2022).

[27] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, X. Wu, Unifying Large Language Models and Knowledge Graphs: A Roadmap, 2023. URL: https://arxiv.org/abs/2306.08302. doi:10.48550/ARXIV.2306.08302.

[28] L. Yang, H. Chen, Z. Li, X. Ding, X. Wu, ChatGPT is not Enough: Enhancing Large Language Models with Knowledge Graphs for Fact-aware Language Modeling, 2023. URL: https://arxiv.org/abs/2306.11489. doi:10.48550/ARXIV.2306.11489.