Temporal Word Embeddings for Narrative Understanding

Claudia Volpetti
Politecnico di Milano
Via Lambruschini 4b
20156 Milan (Italy)
claudia.volpetti@polimi.it

Vani K
IDSIA
Galleria 2 - Via Cantonale
Manno - Lugano (Switzerland)
vanik@idsia.ch

Alessandro Antonucci
IDSIA
Galleria 2 - Via Cantonale
Manno - Lugano (Switzerland)
alessandro@idsia.ch

ABSTRACT

We propose temporal word embeddings as a suitable tool to study the evolution of characters and their sentiments across the plot of a narrative text. The dynamic evolution of instances within a narrative text is a challenging task, where complex behavioral evolutions and other characteristics specific to the narrative text need to be inferred and interpreted. While starting from an existing approach to the learning of these models, we propose an alternative initialization procedure which seems to be especially suited for the case of narrative text. As a validation benchmark, we use the Harry Potter series of books as a challenging case study for such character trait evolution. A benchmark data set based on temporal word analogies related to the characters in the plot of the series is considered. The results are promising, and the empirical validation seems to support the working ideas behind this proposal.

CCS Concepts

• **Artificial intelligence → Natural language processing**
• **Machine learning → Neural networks.**

Keywords

Natural Language Processing; Word Embeddings; Temporal Word Embeddings; Narrative Understanding; Character-Centric Narrative Understanding; Temporal Word Analogies.

1. INTRODUCTION

Narrative Understanding (NU) tasks are natural language understanding techniques specifically designed to process narrative texts and automatically extract from them higher-level information. NU examples are associated to the concepts of narrative storytelling, event chain analysis, narrative generations and inferencing to social media narrative analysis. Efforts in NU are focused on learning the sequence of events by which a story is defined; in this tradition we might situate seminal work on learning procedural scripts [1,2], narrative chains [3], and plot structure [4].

If those works are *story-centric*, i.e., the focus is on the plot of the story, and some other approaches are *author-centric*, i.e., focused instead on plot coherence, here we analyze much more the characters and their relations. Those *character-centric* approaches are focused on character believability, i.e., the extent to which the characters in a story exhibit rich and diverse interactions, emotions, social behavior and motivations [5]. *Character-centric NU* (CNU) tasks are therefore methods focused on understanding and exploring such character believability attributes of the narratives from a social perspective. Topics include identifying characters in narratives, modeling characters as social goal-oriented agents, their interaction with other characters or the environment, their similarity with other entities, their evolution over time, and others. Acting under the CNU umbrella, we consider the task of automatically identifying *character roles and their evolution* over time. A character *role* describes what function a character serves in the story. The character *evolution* is the idea in writing that a character can ideally change from the beginning of a work to the last sentence, e.g., from the villain to the hero. For a first validation of these techniques, we use J.K. Rowling's *Harry Potter* books as a benchmark providing a consistent amount of text, with a story spread over multiple books with recurrent characters and varying relations among them.

Tackling of such a task inherently demands an integration of natural language processing and advanced machine learning techniques. In our approach to CNU, we use *temporal word embeddings* (TWEs), as a tool to represent the time-varying semantic distributions of a vocabulary. A *word embedding* E is a map from a vocabulary V of size v to a d-dimensional real space, i.e., $E: V \to \mathbb{R}^d$, provided together with a metric $\delta: \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, that evaluates the relative distance between vectors. Given two words $w_1, w_2 \in V$, the nonnegative real number $\delta(w_1, w_2)$ measures the dissimilarity level between the two words [6]. Word embedding training is achieved within neural networks architecture. In the simplest setup, a v-dimensional input layer goes to a d-dimensional hidden layer through a $v \times d$ input weight matrix $\boldsymbol{W}$ (also called *word embedding matrix*) and the hidden layer goes to a v-dimensional output layer through a $d \times v$ output weight matrix $\boldsymbol{W'}$ (also called *context matrix*). Each word in a text together with its neighboring word(s) can be used as a set of input/output data able to train the word-to-word map $\boldsymbol{W} \cdot \boldsymbol{W'}$, and $\boldsymbol{W}$ alone eventually provides the required embedding. TWE models are recently proposed approaches to the dynamic learning of word embeddings, i.e., vectors that represent the meaning of words, during a specific temporal interval. Formally, a TWE $\{E_t\}_{t\in T}$ is just a parametrized set of word embeddings, where the parameter t belongs to a set T, that can be discrete or continuous, and for each $t \in T$, E_t is a word embeeding defined as in the previous paragraph. An example is in [7], a TWE is expected to associate different vectors to the word *gay* at different times in the history: its vector in 1900 is expected to be more similar to *cheerful* than its vector in 2005.

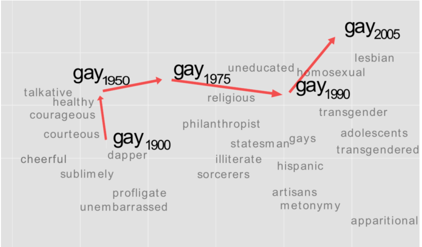

Figure 1 Two-dimensional visualization of semantic changes in English using temporal word vectors of the word *gay* [19].

Stemming from this idea, our claim is that if we construct TWEs (also known as *dynamic word embeddings* [8] or *diachronic word embeddings* [7]) for each character in different time periods (e.g., for each character in each book of a book series) they can be used to represent the role and the evolution of a character along a story plot. TWEs make it possible to find distinct words that share a similar meaning in different periods of time by retrieving temporal embeddings that occupy similar regions in the vector spaces that correspond to distinct time periods. Consequently, our hypothesis is that characters having the same role should be closer, according to some metric measure in the embedding space, to similar characters. E.g., villains should be clustered in a different area from the area in which story heroes are placed. On the other hand, we also claim that by building a sequence of temporal embeddings of a character over consecutive time intervals, one can track the character evolution (semantic shift) occurred in the character role.

Moreover, in this work, we suggest the use of *Temporal Word Analogies* (TWAs) [9] as a tool to evaluate character evolution, since TWAs are one of the standard approaches to the evaluation of TWEs in general. A TWA holds when two words share a common meaning at two different points in time, e.g., "*Ronald Reagan in 1987 is like Bill Clinton in 1997*". The task is therefore to find the word w^* with the semantic role at time t most similar to that of a word w' at a different time t', i.e.,

$$w'\!:t' = w^*\!:t$$

Using TWEs to solve TWAs is based on the implicit idea of an *alignment* the semantic areas in the codomains of the different embeddings associated to a TWE. E.g., an area associated to the *US President* occupied by *Ronald Reagan* vector in 1987 and by *Bill Clinton* vector in 1990. Accordingly, a TWE-based of a TWA is:

$$w^* = \arg\min_{w\in V} \delta(E_t(w), E_{t'}(w'))$$

Accordingly, we can use TWAs to validate the hypothesis that TWEs of characters can be used for CNU as they provide information on characters roles and evolution. In the considered benchmark, the different books of the series are natural timestamps for the TWE and we consider therefore TWA as the following:

Voldemort : Book I = ? : BookII

i.e., who is the character whose role in the second book is more similar to that of Voldemort in the first book? The accuracy in solving such TWAs is therefore a possible proxy of the effectiveness of adopting TWEs in CNU. To measure such accuracies, we create a data set of TWAs across all the books of the Harry Potter saga, gathered through ten annotators with deep knowledge and understanding of these books. To the best of our knowledge, this is the first work that attempts to learn explicit character roles and their evolution in narratives by TWEs.

The paper is organized as follows. Section 2 summarizes the state of the art in CNU. Section 3 discusses the experimental setup with details on TWAs data sets and our approach to TWE training. Section 4 reports on the experimental results. The paper is concluded in Section 5 with brief insights to future directions.

2. RELATED WORK

Automated story understanding is a long-pursued task for AI. This has been approached as a commonsense reasoning task, by which systems make inferences about events that prototypically occur in common experiences. Early works often failed to scale beyond narrow domains of stories due to the difficulty of automatically inducing domain-specific knowledge. The shift to data-driven AI established new opportunities to acquire this knowledge automatically from story corpora. Nowadays natural language processing recognizes that the type of commonsense reasoning used to predict what happens next in a story, for example, is as important for natural language understanding systems as linguistic knowledge itself. Regarding the specific area of CNU, as already mentioned in the introduction, most of the efforts have been in the direction of character identifications and understanding the evolutions on semantic spaces. This includes prediction of event sequences, emotional trajectories [10,11], identification of sentiments and relations [12,13] and generation of character networks and other visualizations [14,15].

3. EXPERIMENTAL FRAMEWORK

We intend to explore the semantic and temporal spaces of characters in the narrative using TWE. In this section, we discuss how we train TWEs and test them using a TWAs data set.

3.1 Training Data

When training TWEs, the amount of information we are able to encode is heavily influenced by the type and size of textual data being used for their training and the temporal granularity of the data [9]. For our experiments we considered as a corpus the six books from the Harry Potter's series. Since the training process of a TWEs relies on diachronic text corpora, we need to decompose our corpus into temporal slices [7,8]. Usually, temporal intervals are set according to the granularity of time spans we want to cover with TWEs [9]. Since we are trying to trace major changes in characters behaviors and role, we decided to keep the granularity of time spans low and consequently we set our *time unit* (the granularity of the temporal dimension) to the number of books. As a result of this choice after the training every word will have six representations, one per each time unit (per each book). Note that we work under the assumption that both the *narrative order* and the *chronological order* of the events and character evolution coincide in the corpus.

Table 1 – Temporal Word Analogies Data Set

Book	Main Antagonist	Second Antagonist	Main Alley	Second Alley	Third Alley
I	*Voldemort*	*Quirrell*	*Ron*	*Hermione*	*Hagrid*
II	*Riddle*	*Basilisk*	*Ron*	*Hermione*	*Hagrid*
III	*Dementors*	*Pettigrew*	*Ron*	*Hermione*	*Lupin*
IV	*Voldemort*	*Crouch*	*Hermione*	*Ron*	*Cedric*
V	*Voldemort*	*Umbridge*	*Ron*	*Hermione*	*Sirius*
VI	*Voldemort*	*Snape*	*Ron*	*Hermione*	*Dumbledore*

3.2 Test Data on Temporal Word Analogies

To test the strength of TWE in CNU we consider a TWAs data set that we built on purpose for this task. This section illustrates how we design and build this dataset. We cope with the Harry Potter's books corpus, and asked ten *experts*, i.e., people who carefully and repeatedly read the six books, to answer a survey. They were asked to answer twelve questions about Harry Potter's characters across the first six books. This approach made it possible to trace a series of 150 characters analogies over time. Table 1 reports examples of characters from the first book and their analogues over the following books as gathered from annotators from which TWA ground truth can be obtained. Each column represents a character role, and the names in that column reports the different characters embodying that role at different points in time (i.e., books).

3.3 Character Identification

Characters are a key element of narrative and so character identification is a necessary preprocessing task for the kind of analysis considered in this paper. Named entity recognition tools such as the classic API of the Stanford *dependency parser* can be used for that. Yet, unlike other kind of texts, in the particular case of NU, the same character might often appear with different *aliases* (e.g., *Harry Potter* as *Harry*, *Ronald Weasley* as *Ron* or *Ronald*). The model we want to develop should clearly work at the character level and the different aliases of a same character need to be regarded as a single word vector. A clustering procedure might be achieved by standard techniques from unsupervised learning techniques (e.g., DBSCAN) with a set of additional heuristic rules.

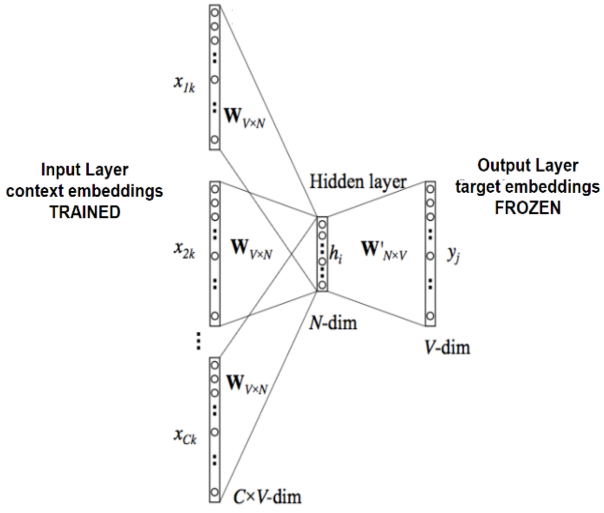

Figure 2 – Training input vectors with frozen output vectors.

3.4 Training Temporal Word Embeddings

Recently, different researchers have been demonstrating the efficiency of tracing temporal changes in lexical semantics using an approach known as distributional models. Such models seem well suited for monitoring the gradual process of meaning change of words over time. Several recent publications demonstrated these models to be efficient and outperform the frequency-based methods in detecting semantic shifts of words over time [7, 17,18,19]. In particular we focus on the case of TWEs.

Many training methods for TWEs suffer from *alignment* issues, i.e., once you train separate word embedding at different time periods (on different corpus slices), it does not make sense to directly calculate similarities between vectors of one and the same word in two different time periods. This is related to the inherent stochasticity of most word embedding training algorithms. To solve this, [19] suggested to first align the models and the calculating similarities. Yet, it has been shown that alignment can compromise the information encoded in the embeddings.

The specific method introduced in [16] seems instead to be able to implicitly align different temporal representations using a shared coordinate system instead of enforcing vector similarity in the alignment process. The same model also proved to be easy to implement on the top of continuous bag of words and skip-grams as Word2vec architecture and highly efficient to train.

This method is built on the assumption that a word, e.g., *Clinton* appears during some temporal periods in the contexts of words that are related to his position, e.g., *president*, that conversely doesn't change its meaning. This assumption allows to heuristically consider the context matrix as static, i.e., to freeze the output weight matrix during training, while allowing the word embedding input weight matrices, to change on the basis of co-occurrence frequencies that are specific to a given temporal interval (Figure 2). After training, model returns the context embeddings, that we are going to consider as a TWE.

This is achieved by a two-fold training procedure. First a static word embedding is trained, with random initialization, using the entire vocabulary and ignoring temporal slices. Let us denote as $\boldsymbol{W}$ the corresponding word embedding matrix and as $\boldsymbol{W'}$ the corresponding context matrix. The word embedding matrices of the TWE, say $\{\boldsymbol{W}_t\}_{t\in T}$, is achieved by initializing these matrices with $\boldsymbol{W}$ and keeping $\boldsymbol{W'}$ as a *frozen* context matrix equal for all the time slices. This initialization has been proved to force alignment and make it possible to compare vectors from embeddings associated to different time slices. Note also that the same procedure with $\boldsymbol{W}$ frozen and $\boldsymbol{W'}$ as initialization could be considered.

In this paper we propose a different initialization scheme for such a training architecture. Having $\boldsymbol{W}$ as the same initialization for all the word embeddings associated to different time slices reflects the idea of a common background of *semantically static* words, which are practically not changing their meaning over time. In the particular case of the characters of a narrative text the situation might be different, as basically each character might change their semantic position over time. For this reason, a better initialization strategy might consist in using $\boldsymbol{W}_{t-1}$as the initialization of $\boldsymbol{W}_t$ and so on, while using $\boldsymbol{W}$ only for the model of the first slice $\boldsymbol{W}_0$. We call this procedure *dynamic initialization*, while the original procedure proposed in [16] is called here *static initialization*. A graphical summary of the architecture together with the two initialization strategies are depicted in Figure 3.

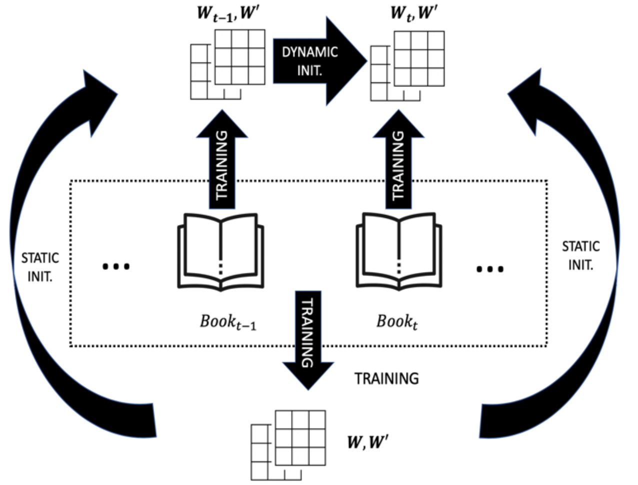
$\boldsymbol{W}_{t-1}, \boldsymbol{W'}$
DYNAMIC INIT.
$\boldsymbol{W}_t, \boldsymbol{W'}$
TRAINING
TRAINING
STATIC INIT.
...
$Book_{t-1}$
$Book_t$
...
STATIC INIT.
TRAINING
TRAINING
$\boldsymbol{W}, \boldsymbol{W'}$

Figure 3 - Temporal context embeddings architecture with both static and dynamic initialization.

Conference'10, Month 1–2, 2010, City, State, Country.

DOI: http://dx.doi.org/10.1145/12345.67890

4. RESULTS AND ANALYSIS

In this section, we discuss the details of our experimental results obtained from the baseline model [16] and the variant with dynamic initialization we proposed in this paper. This is achieved by means of an implementation of the continuous bag of words and the negative sampling extending the Gensim library.[1]

More specifically, we trained both models on the entire Harry Potter corpus (six books) to build the static embeddings and then we trained separately the temporal embeddings according the two different approaches discussed in the previous paragraph. After some hyperparameters tuning procedures, we fixed $d = 200$ for the word embeddings dimensionality, we specified a window size equal to two, and twenty epochs for the static training. Temporal embeddings were trained for five epochs each, when replicating the original approach, and gradually scaled from ten to one when dealing with the second approach.

4.1 Temporal Embeddings Visualization

Semantic trajectories, consisting in the set of vectors corresponding to the same word over different times are the most straightforward product provided by a TWE. Standard techniques such as t-SNE[2] is used to project the high-dimensional vector to spaces of dimension two or three and visualize the semantic trajectories as in the examples in Figure 1 or in Figure 4.

Here are going to plot the temporal embeddings resulting from the training. We are going to discuss the ones produced by the original approach here in the following, since both approaches provided very similar visualizations.

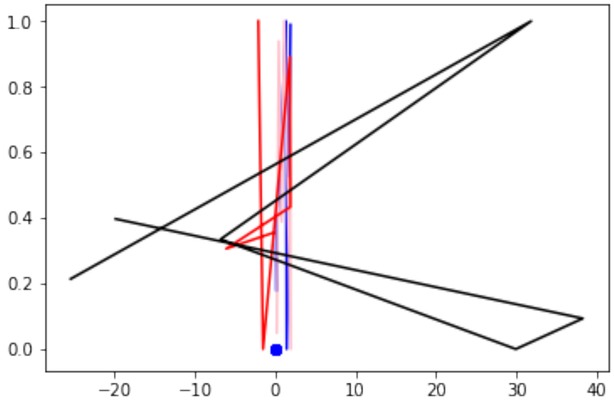

Figure 4 – Semantic trajectories across the six books of five characters: Harry (blue dot), Ron (blue line), Hermione (pink line), Draco (red line) and Voldemort (black line).

In Figure 4, the fixed point is representing Harry and the lines are reporting the behavior through time of the other characters. Hermione and Ron cover the same role in the story plot, and both are main Harry's alleys, figure shows clearly that they both follow the same path, since their lines are close and behave similarly. Voldemort is the main antagonist through books, as it is clearly depicted in figure, its line follows a different path far from the alleys and occupy very different areas in space. Interestingly different is the Draco's vector behavior. Draco character ranges from serving as a secondary antagonist to supporting antagonist, to finally being the central antagonist far ahead in the story plot. This more complex evolution of the character can be confirmed by observing the character evolution line in figure. It is a path which is close to Ron and Hermione behaviors but has a lot of traits of the main antagonist (Voldemort) path.

In summary, Figure 4 seems to demonstrate that TWEs can be used to plot character evolutions and identify character roles by observing their movements and positions in the vector space.

4.2 Temporal Word Analogies Results

From an implementational point of view, solving a TWA is just a matter of retrieving the temporal vector of character in a particular book, and then finding the closer point to that vector among all vectors in a second book as in the equation in Section 1. The resulting vector will be the solution of the analogy. Following this procedure, we can find the characters in a second book most similar to a certain character in another book. Given the TWAs dataset we introduced, we used our models to predict the correct results of 150 temporal character analogies.

Table 2 – Example of Temporal Model Predictions

Antagonist book$_A$	**Antagonist book$_B$**	**Prediction book$_B$**
Voldemort	**Riddle**	**Riddle** ✓
Quirrell	Basilisk	Snape ✕
Ron	**Ron**	**Ron** ✓
Hermione	**Hermione**	**Hermione** ✓
Hagrid	Hagrid	Colin ✕

Accuracy is chosen as metric to count how many correct analogies are predicted. It simply counts the number of correct predictions divided by the total number of analogies. We also provide the accuracy calculated for the *static* and *dynamic* analogies separately (Figure 5). Static analogies involve the same word. E.g., *Voldemort: Book1 = Voldemort: Book3* is static one, while dynamic analogies involve different words.

In Table 2, we report an example of predictions for both static and dynamic analogies. You can interpret the table as follows: e.g. first row is a dynamic analogy since the characters involved are different and the prediction of our model is in this case correct; firth row reports a static analogy and in this case our model output the wrong prediction.

The results of the experiments are summarized in Figure 5. Both models reached very similar performances in terms of general accuracy. We should highlight the fact that both models don't have any difficulty in predicting static analogies (both reach more than 99% of accuracy) and that our variant performs slightly better when facing dynamic analogies.

TWE Model	**Accuracy**	**Static**	**Dynamic**
Static Initialization	65.07 (97.6/150)	99.63 (53.8/54)	45.62 (43.8/96)
Dynamic Initialization	65.14 (97.7/150)	99.26 (53.6/54)	45.94 (44.1/96)

Figure 5 – Accuracy performances on temporal word analogies data set in case of all, only static and only dynamic analogies.

[1] https://github.com/valedica/twec

[2] https://scikit-learn.org/

Finally, in order to be sure to use a fair metric, we calculated a different type of metric as alternative to the one in Figure 5. The accuracy metric used so far, was designed to consider not only the first prediction, but the five top closer vectors predicted as similar characters by using a weighted sum of the errors. In Figure 6, we also provide alternative accuracy results including only the top two predictions. In this case, again, models' performances are comparable, and we record a slightly better outcome for the original model.

TWE Model	**Accuracy (Top 5)**	**Accuracy (Top 2)**
Static Init.	65.07 (97.6/150)	54.8 (82.2/150)
Dynamic Init.	65.14 (97.7/150)	54.3 (81.4/150)

Figure 6 - Accuracy performances on TWAs benchmark.

5. CONCLUSIONS AND OUTLOOKS

We studied temporal word embeddings as a possible tool for effective character-centric narrative understanding. We provided a new data set of temporal word analogies and tested a variant of a recently proposed temporal embedding against it. Results show a good accuracy when solving those character analogies across time. This supports that idea that these embeddings can properly understand the semantic role of each character, the results being particularly robust in case of static analogies. We also provided a visualization of the temporal embeddings to trace the evolution over time of characters in a story plot.

As a future work, we would like to use those embeddings for more CNU tasks and also moves from narratives from social media. An important application of the identification and analysis of such character-centric narratives in social media could be the identification of victims and bullies in hate-speech dialogues.

6. REFERENCES

[1] C. W. Welin, "Scripts, plans, goals and understanding, an inquiry into human knowledge structures: Roger C. Schank and Robert P. Abelson Hillsdale," *J. Pragmatics, Lawrence Erlbaum Assoc.*, vol. 3, no. 2, pp. 211–217, Apr. 1979.

[2] M. Regneri, A. Koller, and M. Pinkal, "Learning script knowledge with web experiments," in *ACL 2010 - 48th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 2010, pp. 979–988.

[3] N. Chambers and D. Jurafsky, "Unsupervised learning of narrative event chains," in *ACL-08: HLT - 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, 2008, pp. 789–797.

[4] M. Mark A. 1977- Finlayson, "Learning narrative structure from annotated folktales," 2012.

[5] Riedl, Mark O., and R. Michael Young. "Character-focused narrative generation for execution in virtual worlds," in *International Conference on Virtual Storytelling*, Springer, Berlin, Heidelberg, 2003, pp. 47-56.

[6] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, 2013, pp. 3111–3119.

[7] W. L. Hamilton, J. Leskovec, and D. Jurafsky, "Diachronic word embeddings reveal statistical laws of semantic change," in *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers*, 2016, vol. 3, pp. 1489–1501.

[8] R. Bamler and S. Mandt, "Dynamic word embeddings," in *34th International Conference on Machine Learning, ICML 2017*, 2017, vol. 1, pp. 607–621.

[9] T. Szymanski, "Temporal Word Analogies: Identifying Lexical Replacement with Diachronic Word Embeddings," pp. 448–453.

[10] Chaturvedi, S., Peng, H. and Roth, D., 2017, September. Story comprehension for predicting what happens next. *In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* , pp. 1603-1614.

[11] Vani, K. and Antonucci, A., 2019. NOVEL2GRAPH: Visual Summaries of Narrative Text Enhanced by Machine Learning. *In Text2Story@ ECIR,* pp. 29-37.

[12] Nalisnick, E.T. and Baird, H.S., 2013, August. Character-to-character sentiment analysis in shakespeare's plays. *In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics,* vol.2, pp. 479-483.

[13] John, M., Lohmann, S., Koch, S., Wörner, M., and Ertl, T., Visual Analytics for Narrative Text, 2016.

[14] Labatut, V., and Bost, X., 2019. Extraction and Analysis of Fictional Character Networks: A Survey. *arXiv preprint arXiv:1907.02704*.

[15] Roemmele, M., and Gordon, A, "An Encoder-decoder Approach to Predicting Causal Relations in Stories." *Proc. of the First Workshop on Storytelling*, pp. 50-59, 2018.

[16] V. Di Carlo, F. Bianchi, and M. Palmonari, "Training Temporal Word Embeddings with a Compass," *Proc. AAAI Conf. Artif. Intell.*, vol. 33, pp. 6326–6334, Jul. 2019.

[17] A. Kutuzov, L. Øvrelid, T. Szymanski, and E. Velldal, "Diachronic word embeddings and semantic shifts: a survey," 2018.

[18] Z. Yao, Y. Sun, W. Ding, N. Rao, and H. Xiong, "Dynamic word embeddings for evolving semantic discovery," in *WSDM 2018 - Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, 2018, vol. 2018-Febua, pp. 673–681.

[19] V. Kulkarni, R. Al-Rfou, B. Perozzi, and S. Skiena, "Statistically significant detection of linguistic change," in *WWW 2015 - Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 625–635.

Authors' background

Your Name	Title*	Research Field	Personal website
Claudia Volpetti	Ph.D. Candidate	AI, NLP, Machine Learning	
Vani K	Lecturer	AI, NLP, Machine Learning	
Alessandro Antonucci	Senior Lecturer	AI, Statistical Machine Learning	idsia.ch/~alessandro