

# SCORE: STORY COHERENCE AND RETRIEVAL ENHANCEMENT FOR AI NARRATIVES

*Qiang Yi<sup>1\*</sup>, Yangfan He<sup>2\*</sup>, Jianhui Wang<sup>3\*</sup> Xinyuan Song<sup>4</sup>, ShiYao Qian<sup>5</sup>, Xinhang Yuan<sup>6</sup>, Yi Xin<sup>7</sup>, Yijin Wang<sup>8</sup>, Jingqun Tang<sup>9</sup>, Yuchen Li<sup>10</sup>, Junjiang Lin<sup>17</sup> Hongyang He<sup>11</sup>, Zhen Tian<sup>12</sup>, Tianxiang Xu<sup>19</sup>, Keqin Li<sup>13</sup>, Kuan Lu<sup>14</sup>, Menghao Huo<sup>15</sup>, Jiaqi Chen<sup>14</sup>, Miao Zhang<sup>16</sup>, Tianyu Shi<sup>5</sup>, Jianyuan Ni<sup>18†</sup>*

<sup>1</sup> UCB, <sup>2</sup> UMN, <sup>3</sup> UESTC, <sup>4</sup> Emory, <sup>5</sup> UofT, <sup>6</sup> WUSTL, <sup>7</sup> NJU, <sup>8</sup> XDU, <sup>9</sup> ByteDance, <sup>10</sup> Baidu, <sup>11</sup> UofWarwick, <sup>12</sup> UofGlasgow, <sup>13</sup> AMA, <sup>14</sup> Google, <sup>15</sup> SCU, <sup>16</sup> THU-SZ, <sup>17</sup> Amazon, <sup>18</sup> JC, <sup>19</sup> PKU

## ABSTRACT

Large Language Models (LLMs) can generate creative and engaging narratives from user-specified input, but maintaining coherence and emotional depth throughout these AI-generated stories remains a challenge. In this work, we propose SCORE, a framework for Story Coherence and Retrieval Enhancement, designed to detect and resolve narrative inconsistencies. By tracking key item statuses and generating episode summaries, SCORE uses a Retrieval-Augmented Generation (RAG) approach to identify related episodes and enhance the overall story structure. Experimental results from testing multiple LLM-generated stories demonstrate that SCORE significantly improves the consistency and stability of narrative coherence compared to baseline GPT models, providing a more robust method for evaluating and refining AI-generated narratives.

**Index Terms**— Large language models, AI narrative, story structure.

## 1. INTRODUCTION

Deep learning has transformed multiple domains including NLP, time series analysis and computer vision [1, 2, 3, 4, 5, 6]. Large Language Models (LLMs) have demonstrated significant capabilities in generating long-form narratives, such as serialized stories or novels, by leveraging large-scale architectures and vast amounts of training data [7, 8, 9]. However, maintaining narrative consistency over extended texts, especially in terms of character development and emotional coherence, remains a major challenge [10]. For instance, [11] pointed out that achieving thematic consistency and managing dynamic plot states is crucial for maintaining the logical flow of a story. In practice, LLMs often struggle with inconsistencies when characters or key plot items reappear without proper explanation, disrupting the overall narrative structure [12].

Similarly, [13] highlight the difficulties in managing multimodal elements within long-form narratives, noting that inconsistencies in character behavior or emotional tone can negatively impact reader engagement. These challenges indicate

a need for more structured approaches in narrative generation that can better manage character arcs, plot developments, and emotional progression throughout the story.

In addition, recent works have highlighted the importance of memory mechanisms in LLM-based agents. [14] conducted a comprehensive survey on these mechanisms, identifying effective memory designs that help mitigate inconsistencies in narrative development—a challenge common to both interactive agents and narrative generation tasks. Additionally, [15] introduced generative agents that simulate human-like behavior using memory modules. These agents track the state of a wide array of interactable objects in a sandbox environment, ensuring consistent reasoning and enabling the smooth functioning of a simulated society. These researches inspired the design of our new framework.

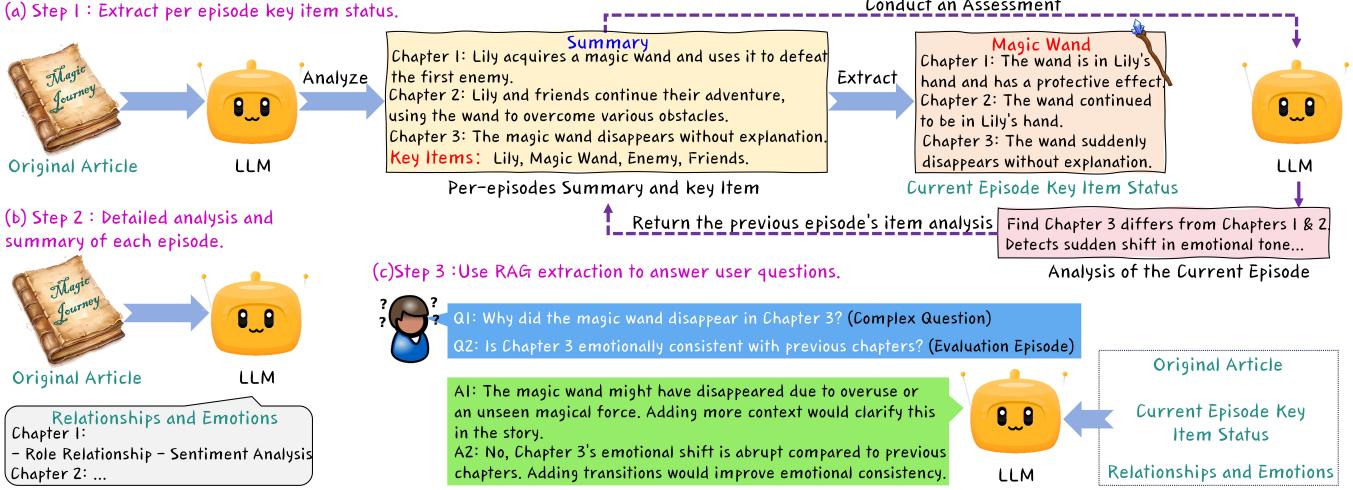
In this work, we build upon recent advancements in Retrieval-Augmented Generation (RAG) [16], which dynamically incorporates relevant context to enhance narrative coherence. Expanding on these developments, we propose SCORE, a framework designed to evaluate three critical aspects of long-form narrative generation: character consistency, emotional coherence, and logical tracking of key plot elements. Our key contributions are:

- We introduce SCORE, an LLM-based evaluation framework that detects narrative inconsistencies in AI-generated stories.
- We incorporate a Retrieval-Augmented Generation (RAG) approach, utilizing episode-level summaries and key item tracking to improve narrative coherence.
- We demonstrate enhanced story consistency and emotional depth by integrating sentiment analysis and similarity-based episode retrieval. Specially, we outperform baseline GPT model [17] in detecting continuity errors and maintaining overall narrative coherence.

## 2. METHOD

Our proposed method, SCORE, consists of three main components: (1) an LLM-based evaluation framework to assess

\* Equal contribution. † Corresponding author.



**Fig. 1: Overview of SCORE:** (a) Extracts key item statuses per episode. (b) Conducts detailed analysis and summaries of each episode. (c) Uses RAG to answer user queries and resolve narrative inconsistencies.

the coherence of key story elements, (2) automatic generation of episode summaries to track plot development, and (3) a retrieval-augmented generation (RAG) approach that integrates the first two components, enabling enhanced user interaction and ensuring narrative consistency.

As the framework is intended solely for academic research purposes, its use is consistent with the original access conditions of all incorporated tools and data sources.

## 2.1. Continuity Analysis and Key Item Status Correction

By extracting key parts of GPT-4’s analysis, we identify instances where an item reappears in later episodes after being marked as lost or destroyed, without narrative explanation. Let  $S_i(t)$  represent the state of item  $i$  at time  $t$ , where  $S_i(t) \in \{\text{active}, \text{lost}, \text{destroyed}\}$ . If item  $i$  reappears at time  $t_k$  with  $S_i(t_k) = \text{active}$  after being previously marked as  $S_i(t_{k-1}) \in \{\text{lost}, \text{destroyed}\}$ , we flag this as a continuity error. To maintain consistency, the state remains  $S_i(t_{k-1})$ , avoiding an incorrect update. This approach systematically corrects discrepancies in item states, ensuring that narrative continuity is preserved by preventing erroneous state transitions.

## 2.2. Key Item Interaction Analysis

For each episode, we conduct a thorough evaluation by summarizing key plot points, character actions, and tracking interactions with important items. Let  $A_c(t)$  represent the actions of character  $c$  at time  $t$ , and let  $I_i(t)$  denote interactions with key item  $i$ . The model generates summaries that encapsulate essential elements, including  $A_c(t)$  (character actions), relationships, and emotional changes across the episode. It then analyzes the specific interactions  $I_i(t)$  between characters and key items, documenting these for further analysis. This step aggregates relevant narrative information—combining

episode summaries, key item interactions  $I_i(t)$ , and character actions  $A_c(t)$ —to facilitate more precise future retrieval. The approach simplifies subsequent analysis of plot and item continuity, reducing redundancy and improving efficiency.

## 2.3. Similarity-Based Episode Evaluation and Sentiment Analysis

We integrate similarity-based retrieval and sentiment analysis to improve episode evaluation and answer complex queries. It begins by loading summaries, full episode content, and key item states from structured JSON files. The content is segmented into smaller chunks using a text segmenter and embedded into a vector space model using FAISS [18] and OpenAI embeddings. This vector space enables efficient retrieval of similar episodes for user queries or specific episode analysis.

For evaluation, the system retrieves relevant past episodes by calculating cosine similarity scores [19] between the current episode or query and all other episodes in the vector space. Let  $S(e_c, e_p)$  represent the similarity score between the current episode  $e_c$  and a past episode  $e_p$ . The top  $N$  episodes with the highest  $S(e_c, e_p)$  scores are retrieved for further analysis, providing a relevant summary of episodes for evaluation or answering questions.

Sentiment analysis is then applied to both the current and retrieved episodes. A sentiment score  $\sigma(e)$ , ranging from 0 to 1, is assigned to each episode  $e$  by GPT-4, reflecting its emotional tone. These scores help refine the selection by ensuring both text similarity and sentiment consistency are considered, thus preventing errors from large sentiment discrepancies.

Finally, the LLM processes the retrieved episode summaries and content to generate a detailed evaluation. The

**Table 1:** Performance Comparison of LLMs with/without SCORE Framework

Model	Consistency	Coherence	Item Status	Complex QA
GPT-4	83.21	84.32	0	82.34
SCORE	<b>85.61 (↑2.4)</b>	<b>86.9 (↑2.58)</b>	<b>98 (↑98)</b>	<b>89.45 (↑7.11)</b>
GPT-4o	86.78	82.21	0	76.32
SCORE	<b>88.68 (↑1.9)</b>	<b>89.91 (↑7.7)</b>	<b>96 (↑96)</b>	<b>88.75 (↑12.43)</b>
Claude3	84.6	80.9	0	69.45
SCORE	<b>87.2 (↑2.6)</b>	<b>85.7 (↑4.8)</b>	<b>93.1 (↑93.1)</b>	<b>83.9 (↑14.45)</b>
Gemini-Pro	82.2	83.4	0	78.40
SCORE	<b>85.2 (↑3.2)</b>	<b>86.0 (↑3.0)</b>	<b>95.0 (↑95.0)</b>	<b>85.5 (↑7.10)</b>
Qwen-14B	79.3	76.2	0	25.15
SCORE	<b>84.5 (↑5.2)</b>	<b>79.1 (↑2.9)</b>	<b>86.3 (↑86.3)</b>	<b>61.2 (↑36.05)</b>
GPT-4o-mini	77.5	75.8	0	23.40
SCORE	<b>81.9 (↑4.4)</b>	<b>76.6 (↑0.8)</b>	<b>79.8 (↑79.8)</b>	<b>60.1 (↑36.7)</b>
Llama-13B	71.3	69.8	0	18.72
SCORE	<b>79.1 (↑7.8)</b>	<b>73.4 (↑3.6)</b>	<b>76.2 (↑76.2)</b>	<b>57.3 (↑38.58)</b>

focus is on narrative aspects such as character consistency, plot progression, emotional authenticity, and key item continuity. This ensures the narrative remains coherent, with any discrepancies flagged and corrected.

### 3. EXPERIMENTS

To evaluate the effectiveness of SCORE, we conducted experiments on stories generated by large language models (LLMs). These experiments assessed the framework’s ability to maintain narrative coherence, detect continuity errors, and ensure emotional consistency throughout episodic storytelling.

#### 3.1. Dataset Preparation

We constructed a story dataset of 5,000 episodes from 1,000 GPT-generated stories evenly distributed across four genres (science fiction, drama, fantasy, comedy). Each story contains 10–15 episodes (avg. 12, ~2,000 tokens), generated using GPT-3.5 and GPT-4 outputs with 15 diverse prompt templates per genre. Quality was ensured via manual filtering by three annotators ( $\kappa=0.78$ ), and complexity stratification included multi-character interactions (30%), non-linear timelines (20%), and symbolic systems (50%). For broader validation, SCORE was also tested on NarrativeQA, BookCorpus, and WP-STORIES, with genre and episode length distributions standardized for cross-dataset comparison. This hybrid setup balances synthetic and human-authored narratives, ensuring robustness across industrial and academic benchmarks.

#### 3.2. Metrics

We evaluated the framework using several key metrics: **narrative coherence**, assessed by examining logical consistency in

character behavior and plot development to avoid continuity errors; **consistency**, defined as the fraction of responses not conflicting with prior information; **coherence**, measured as the average semantic or logical flow score across responses; **item status**, calculated as the proportion of required elements present in each response; and **complex question**, defined as the fraction of correctly answered complex questions. Finally, we compared the evaluation scores from different methods to measure the stability of the framework.

#### 3.3. Evaluation Process

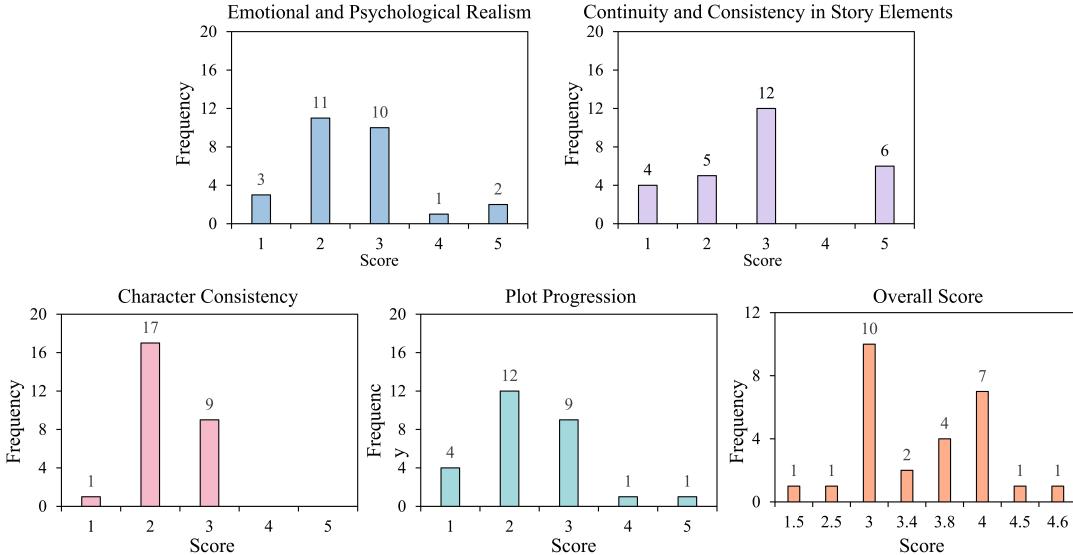
For each episode, evaluation proceeded in two stages. First, we directly uploaded files to ChatGPT (GPT-4o-mini, GPT-4o, GPT-4) to test baseline narrative assessment without prompts. Second, we performed detailed evaluation using preprocessed files with RAG support, where FAISS-based cosine similarity retrieved semantically and emotionally aligned episodes to provide context. This enriched context was then incorporated into GPT prompts for more accurate key-item tracking and narrative evaluation.

#### 3.4. Baselines

We compared our proposed framework to three baselines: GPT-4, GPT-4o, and GPT-4o-mini. In these cases, we used these models directly without integrating our key item tracking, continuity analysis, or retrieval-augmented generation (RAG) mechanisms. We used the same LLM-generated stories to evaluate different models. For all baselines, we measured their ability to evaluate episodes independently and answer complex questions, without deliberately guiding them through the details of the story.

**Table 2:** Ablation results for our complete configuration (Full SCORE) versus removing each module.

Configuration	Consistency	Coherence	Item Acc.	Complex QA
Full SCORE	89.7	91.2	98.3	88.5
w/o Dynamic Tracking	72.1 ( $\downarrow 17.6$ )	85.4 ( $\downarrow 5.8$ )	61.2 ( $\downarrow 37.1$ )	67.3 ( $\downarrow 21.2$ )
w/o Context Summary	83.2 ( $\downarrow 6.5$ )	68.7 ( $\downarrow 22.5$ )	89.1 ( $\downarrow 9.2$ )	71.4 ( $\downarrow 17.1$ )
w/o Hybrid Retrieval	86.4 ( $\downarrow 3.3$ )	88.9 ( $\downarrow 2.3$ )	94.2 ( $\downarrow 4.1$ )	77.6 ( $\downarrow 10.9$ )
w/o Sentiment	87.1 ( $\downarrow 2.6$ )	89.5 ( $\downarrow 1.7$ )	96.8 ( $\downarrow 1.5$ )	82.3 ( $\downarrow 6.2$ )



**Fig. 2:** Case study.

### 3.5. Main Results

As shown in Table 1, our experiments demonstrated that the proposed framework significantly improved the detection of narrative inconsistencies. Evaluations using the framework were able to more accurately detect inconsistencies in character actions or plot progression. The retrieval-augmented generation process helped GPT better filter irrelevant information, understand the current story context, and improved its ability to detect narrative continuity across multiple episodes. When quantitatively compared to baseline methods, such as using GPT model alone, the proposed framework showed substantial improvements in evaluation accuracy.

### 3.6. Ablation Studies

**LLM type.** Table 1 shows that SCORE improves performance across all model families, with especially large gains for open-source models. Consistency increases up to 7.8% (Llama-13B), coherence up to 7.7% (GPT-4o), and item status recognition approaches 98%. Open-source models (e.g., Qwen-14B, Llama-13B) see the largest QA gains (30–38 points), while commercial models (e.g., GPT-4, Claude 3) also benefit, though more modestly. Notably, SCORE boosts GPT-4o beyond GPT-4 in coherence, underscoring its scalability.

**Configuration analysis.** Table 2 shows that Dynamic Track-

ing is critical, with its removal causing the largest drops in consistency (−17.6 points) and item accuracy (−37.1 points). Context Summary is similarly vital for coherence (−22.5%). Hybrid Retrieval and Sentiment contribute positively but with smaller impacts.

### 3.7. Case Study

We generated several dozen stories with SCORE, averaging 555 tokens and 13.2s runtime per episode. Evaluation across four metrics—character consistency, plot progression, emotional realism, and continuity—yielded average scores of 3–4/5 (Figure 2), indicating generally cohesive narratives.

## 4. CONCLUSION

We present SCORE, a novel LLM-based framework that enhances long-term coherence and emotional consistency in AI narratives. By combining Dynamic State Tracking, Context-Aware Summarization, and Hybrid Retrieval within a RAG pipeline, SCORE achieves substantial improvements in coherence, stability, and hallucination reduction across multi-genre datasets. Its modular design ensures scalability and multi-LLM compatibility, though challenges remain in retrieval accuracy and computational efficiency.

## 5. REFERENCES

- [1] Xiangfei Qiu, Xiuwen Li, Ruiyang Pang, Zhicheng Pan, Xingjian Wu, Liu Yang, Jilin Hu, Yang Shu, Xuesong Lu, Chengcheng Yang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, and Bin Yang, “Easymore: Time series forecasting made easy,” in *ICDE*, 2025.
- [2] Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang, “Duet: Dual clustering enhanced multivariate time series forecasting,” in *SIGKDD*, 2025, pp. 1185–1196.
- [3] Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang, “Tfb: Towards comprehensive and fair benchmarking of time series forecasting methods,” in *Proc. VLDB Endow.*, 2024, pp. 2363–2377.
- [4] Shutao Li, Bin Li, Bin Sun, and Yixuan Weng, “Towards visual-prompt temporal answer grounding in instructional video,” *TPAMI*, vol. 46, no. 12, pp. 8836–8853, 2024.
- [5] Bin Li and Hanjun Deng, “Bilateral personalized dialogue generation with contrastive learning,” *Soft Computing*, vol. 27, no. 6, pp. 3115–3132, 2023.
- [6] Bin Li, Bin Sun, Shutao Li, Encheng Chen, Hongru Liu, Yixuan Weng, Yongping Bai, and Meiling Hu, “Distinct but correct: generating diversified and entity-revised medical response,” *Science China Information Sciences*, vol. 67, no. 3, pp. 132106, 2024.
- [7] Meiling Tao, Liang Xuechen, Tianyu Shi, Lei Yu, and Yiting Xie, “Rolecraft-glm: Advancing personalized role-playing in large language models,” in *Proceedings of the 1st Workshop on Personalization of Generative AI Systems (PERSONALIZE 2024)*, 2024, pp. 1–9.
- [8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al., “Language models are few-shot learners,” *NeurIPS*, vol. 33, pp. 1877–1901, 2020.
- [9] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al., “Llama: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971v1*, 2023.
- [10] Dan P McAdams, “The problem of narrative coherence,” *Journal of constructivist psychology*, vol. 19, no. 2, pp. 109–125, 2006.
- [11] Aisha Khatun and Daniel G Brown, “Assessing language models’ worldview for fiction generation,” *arXiv preprint arXiv:2408.07904*, 2024.
- [12] Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao, “Evaluation of text generation: A survey,” *ArXiv preprint*, vol. abs/2006.14799v2, 2020.
- [13] Danyang Liu, Mirella Lapata, and Frank Keller, “Generating visual stories with grounded and coreferent characters,” *arXiv preprint arXiv:2409.13555*, 2024.
- [14] Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen, “A survey on the memory mechanism of large language model based agents,” *arXiv preprint arXiv:2404.13501*, 2024.
- [15] Joon Sung Park, Joseph O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein, “Generative agents: Interactive simulacra of human behavior,” in *UIST*, 2023, pp. 1–22.
- [16] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al., “Retrieval-augmented generation for knowledge-intensive nlp tasks,” *NeurIPS*, vol. 33, pp. 9459–9474, 2020.
- [17] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al., “Language models are unsupervised multitask learners,” *OpenAI blog*, vol. 1, no. 8, pp. 9, 2019.
- [18] Matthijs Douze, Alexandre Guizhava, Chengqi Deng, Jeff Johnson, Gergely Szilvassy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou, “The faiss library,” *arXiv preprint arXiv:2401.08281*, 2024.
- [19] Faisal Rahutomo, Teruaki Kitasuka, Masayoshi Aritsugi, et al., “Semantic cosine similarity,” *ICAST*, vol. 4, no. 1, pp. 1, 2012.