NASH: Numerically Aware Scoring Heuristic for Robust Semantic Similarity

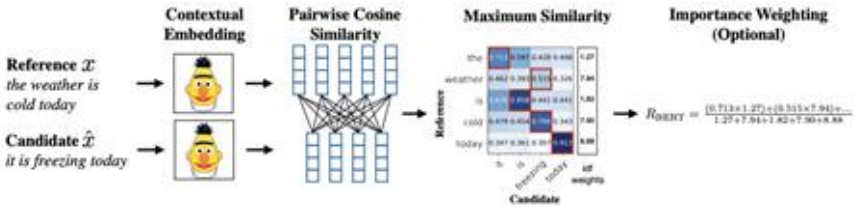
ACL 2026 Findings #3387

Yu-Shiang Huang[*1,3], Yun-Yu Lee[*2], Tzu-Hsin Chou[1], Che Lin[1], Chuan-Ju Wang[3]
[1]National Taiwan University, [2]National Yang Ming Chiao Tung University, [3]Academia Sinica
* These authors contributed equally to this work.

Contact: F09946004@ntu.edu.tw

The Problem: Numerical Blindness

Background:
Embedding-based metrics (e.g., BERTScore) are state-of-the-art for general semantics similarity quantification.



Critical Limitation:
They exhibit a critical limitation: Numerical Blindness in quantitative domains.

Numerical Blindness

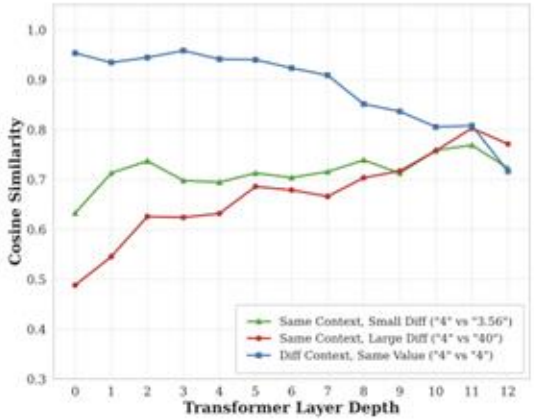
- s_1: *"Revenue increased by **3.56%**."*
- s_2: *"Revenue increased by **4%**."*
- s_3: *"Revenue increased by **40%**."*

- The Paradox: Models map distinct numerical tokens to nearly identical vectors due to shared context.
- Result: sim(s1, s2) = 0.9639 < sim(s2, s3) = 0.9764. with Huggingface BERTScore implementation

Root Cause of the Failure

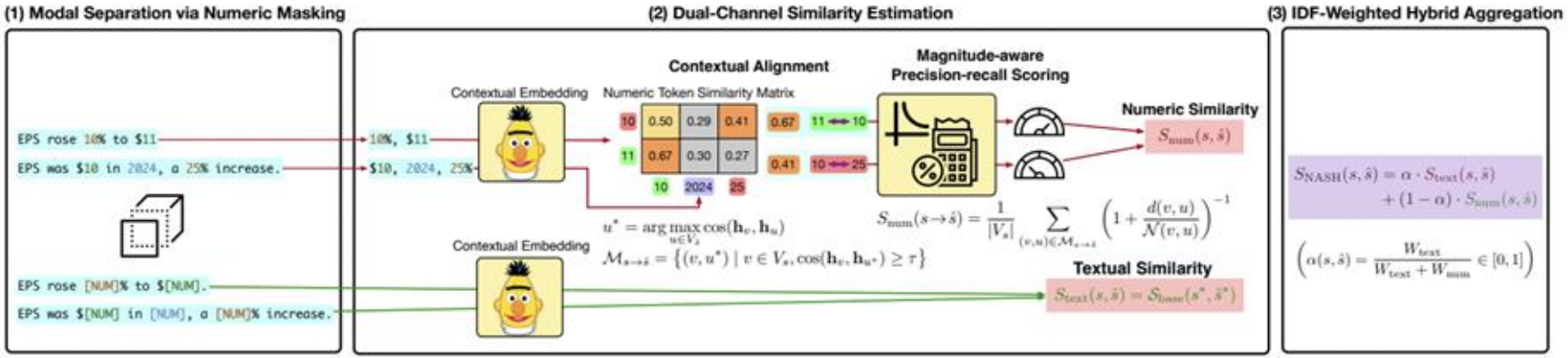
- **Tokenization Fragmentation**
  - Subword tokenizers fragment numbers into disjoint tokens, destroying their inherent magnitude information.
  - Example: The value "3.56" is split into multiple tokens, but "40" remains atomic.
  - This fragmentation prevents systematic magnitude comparison.
- **Contextual Anisotropy**
  - Numerical embeddings are distorted, prioritizing syntactic context over objective numerical content.
  - In identical contexts, embeddings converge rapidly regardless of the magnitude difference.
  - **Crucial Insight:** Paradoxically, the contradictory value (e.g., "40") becomes more similar to the anchor than the numerically closer value (e.g., "3.56") in deeper layers.

$$s_1 : [\ldots, 3, ., 56, \%, .]$$
$$s_2 : [\ldots, 4, \%, .]$$
$$s_3 : [\ldots, 40, \%, .]$$



Overview of NASH

Our Solution - NASH: Numerically Aware Scoring Heuristic for Robust Semantic Similarity

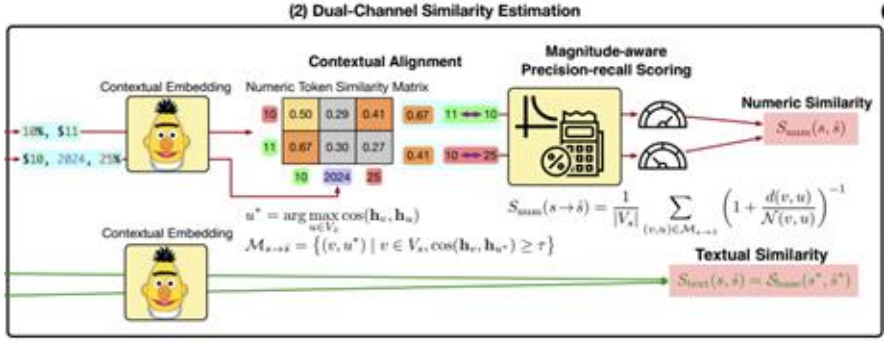


Stage 1 - Modal Seperation

- Goal: We address Contextual Anisotropy by decoupling numerical verification from textual semantics.
- Mechanism:
  Numeric mentions are identified via a rule-based extractor and replaced with a special `[NUM]` token.
- This **Numeric Masking** isolates the textual modality from numerical magnitudes, enabling pure semantic comparison.

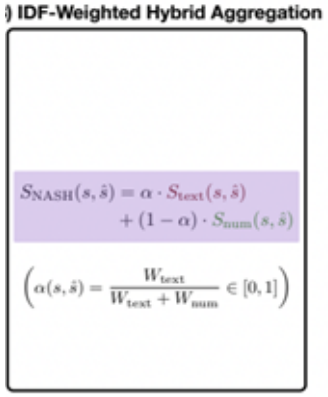
Stage 2 - Dual-Channel Similarity Estimation

1. **Textual Similarity**
   - Assess linguistic similarity independent of numerical values: Compute on the masked sentences.
2. **Context-Aware Numeric Similarity**
   - Explicitly model quantitative relations based on magnitude.
   - **Contextual Alignment:** Leverage contextual embeddings to align numbers serving compatible semantic roles.
   - **Magnitude-aware Scoring:** Aggregate bidirectional scores using a Precision-Recall framework.
     - Penalizes unaligned numbers (low recall).
     - Score is based on the absolute magnitude difference between aligned pairs.



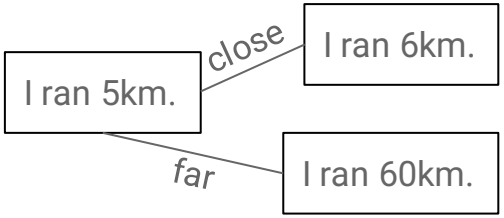
Stage 3 - IDF-Weighted Aggregation

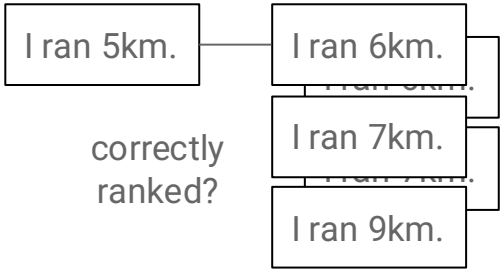
- **Goal: Dynamic Balancing**
  - We dynamically balance the contributions of the Textual and Numerical channels.
  - Higher numerical density results in a lower α (more weight on the numerical channel).
- **Mechanism: IDF Coefficient**
  - The IDF-weighted coefficient is computed based on the ratio of textual mass to total token mass.
- **Final Score:**
  - The final score is a convex combination of both channels
  - When numerical information is absent, NASH gracefully reduces to the textual baseline.

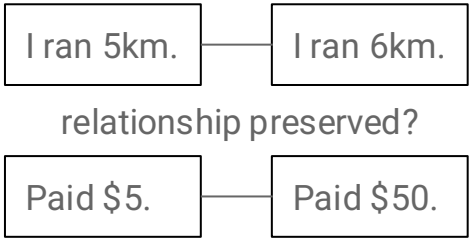


The NumFinE Benchmark

- **Motivation:** Traditional benchmarks (e.g., STS-B) rely on subjective human annotations that fail to expose numerical blindness or enforce axiomatic numerical accuracy.
- **Construction:** We construct NumFinE, a diagnostic benchmark built by applying **controlled numerical perturbation** to 9,227 authentic financial sentences.
    - Sources include Financial News (**Financial PhraseBank**), Social Media (**FinNum1**), and Official Filings (**10-K**).
- **Evaluation Protocols:** We design two protocols to systematically assess numerical sensitivity:
    - **Anchor-based Evaluation:** Compares augmented variants against their base sentence via Triplet and Listwise ranking. Difficulty increases as the required level of fine-grained numerical distinction tightens.
    - **Cross-Pair Evaluation:** Tests whether numerical distance relationships generalize across different sentence contexts.







Main Results

- NASH functions as a drop-in enhancement, consistently boosting performance across all tested embedding architectures in Triplet Setting.
- NASH Achieved substantial gains in numerical sensitivity, with the highest relative improvement of up to **+159.6%** in Listwise ranking
- Effectiveness increases with task difficulty, with improvements widening from Easy (e.g., +7.0%) to Hard Triplet evaluation (up to **+29.5%**).
- Even modern and large-scale embedding models suffer from numerical blindness, validating NASH's general necessity.
- Future work is directed toward improving cross-context normalization mechanisms for robust performance on heterogeneous sentence pairs under cross-pair evaluation.

Model	Triplet (Easy)		Triplet (Med)		Triplet (Hard)		Listwise	
	Base	+NASH	Base	+NASH	Base	+NASH	Base	+NASH
Token-level Encoders								
bert-base-uncased	**0.9214**	0.9857 ↑7.0%	0.8359	0.9745 ↑16.6%	**0.7058**	0.8821 ↑25.0%	0.5409	0.7834 ↑44.8%
FinBERT	0.9186	0.9795 ↑6.6%	**0.8371**	0.9602 ↑14.7%	0.7017	0.8647 ↑23.2%	0.5344	0.7514 ↑40.6%
DeBERTa-xlarge-mnli	0.8715	**0.9859** ↑13.1%	0.7903	**0.9774** ↑23.7%	0.6879	**0.8906** ↑29.5%	**0.6286**	**0.8028** ↑27.7%
Sentence-level Bi-encoders								
all-MiniLM-L6-v2	0.8440	0.9743 ↑15.4%	0.7982	0.9577 ↑20.0%	0.6782	0.8633 ↑27.3%	0.3450	0.7289 ↑111.3%
unsup-simcse-bert-base-uncased	0.7568	0.9301 ↑22.9%	0.7676	0.9038 ↑17.7%	0.7028	0.8095 ↑15.2%	0.2265	0.5879 ↑159.6%
bge-m3	0.8335	0.9325 ↑11.9%	0.7993	0.9093 ↑13.8%	0.6809	0.8230 ↑20.9%	0.3481	0.6065 ↑74.2%
Qwen3-Embedding-0.6B	0.8408	0.9802 ↑16.6%	0.8061	0.9627 ↑19.4%	0.6791	0.8748 ↑28.8%	0.3548	0.7677 ↑116.4%
text-embedding-3-small	0.8759		0.8247		0.6899		0.4111	

Table 2: **Evaluation on NumFinE (Anchor-based).** Results show the original backbone (Base) versus the NASH-enhanced performance (+NASH). Subscripts denote relative improvement percentages. **Bold** indicates best performance in each column.

Model	Cross-Pair Accuracy	
	Base	+NASH
Token-level Encoders		
bert-base-uncased	0.6727	0.6410 ↓4.7%
FinBERT	0.6772	0.6354 ↓6.2%
DeBERTa-xlarge-mnli	0.4715	0.6556 ↑39.1%
Sentence-level Bi-encoders		
all-MiniLM-L6-v2	0.5984	0.6404 ↑7.0%
unsup-simcse-bert-base-uncased	0.6564	0.6330 ↓3.6%
bge-m3	0.5798	0.6143 ↑6.0%
Qwen3-Embedding-0.6B	0.6015	0.6398 ↑6.3%
text-embedding-3-small	0.5892	

Table 5: **Evaluation on NumFinE (Cross-Pair).** Results show the original backbone (Base) versus the NASH-enhanced performance (+NASH). Subscripts denote relative improvement percentages. **Bold** indicates best performance in each column.

Zero-Shot Cross-Domain Generalization

- NASH preserves general semantic performance on standard semantic benchmarks (STS-B).
- It yields consistent precision gains on the domain-specific Financial-STS benchmark.
- The metric demonstrates strong zero-shot cross-domain generalization, achieving substantial improvements on biomedical texts (PubMedQA), evaluated using the same Triplet and Listwise perturbation protocols as NumFinE.
- For more qualitative examples, please refer to our paper appendix.

Model	STS-B (Spearman)		STS-B (Pearson)		Fin-STS (ROC)		Fin-STS (PR)	
	Base	+NASH	Base	+NASH	Base	+NASH	Base	+NASH
Token-level Encoders								
bert-base-uncased	0.4567	0.5007 ↑9.6%	0.4507	0.5086 ↑12.9%	0.6799	0.6849 ↑0.7%	0.6496	0.6519 ↑0.4%
FinBERT	0.5587	0.5720 ↑2.4%	0.5330	0.5779 ↑8.4%	0.6796	0.6833 ↑0.5%	0.6541	0.6554 ↑0.2%
DeBERTa-xlarge-mnli	0.5930	0.5123 ↓12.1%	0.5356	0.5167 ↓3.5%	0.6739	0.6726 ↓0.2%	0.6384	0.6350 ↓0.5%
Sentence-level Bi-encoders								
all-MiniLM-L6-v2	0.8293	0.8235 ↓0.8%	0.8383	0.8342 ↓0.6%	0.6184	0.6282 ↑1.6%	0.5707	0.6036 ↑5.8%
unsup-simcse-bert-base-uncased	0.7815	0.7652 ↓2.1%	0.7915	0.7791 ↓2.0%	0.6547	0.6794 ↑3.8%	0.5958	0.6485 ↑8.9%
bge-m3	0.8467	0.8457 ↓0.1%	0.8404	0.8399 ↓0.1%	0.6529	0.6766 ↑3.6%	0.5850	0.6408 ↑9.5%
Qwen3-Embedding-0.6B	0.4889	0.4904 ↑0.3%	0.5049	0.4936 ↓2.2%	0.6113	0.6616 ↑8.2%	0.5789	0.6074 ↑4.9%
text-embedding-3-small	0.8462		0.8609		0.631		0.5522	

Table 3: **Evaluation on STS-B and Financial-STS.** Results show the original backbone (Base) versus the NASH-enhanced performance (+NASH). Subscripts denote relative improvement percentages. **Bold** indicates best performance in each column.

Model	Triplet (Easy)		Listwise	
	Base	+NASH	Base	+NASH
Token-level Encoders				
bert-base-uncased	0.7080	0.7740	0.4483	0.5453
FinBERT	0.6680	0.7050	0.3887	0.4185
Sentence-level Bi-encoders				
all-MiniLM-L6-v2	0.5440	0.7060	0.1303	0.4187

Table 4: **Zero-shot evaluation on biomedical data constructed from PubMedQA.** NASH is applied completely out-of-the-box without domain-specific tuning and substantially improves numerical sensitivity in biomedical text.

Conclusion

In conclusion, in this paper, we…

- Formally identified numerical blindness in embedding-based metrics as a critical limitation.
- Proposed NASH, decoupling textual semantics from explicit numerical verification.
- Implemented a three-stage pipeline: masking, alignment, and IDF aggregation.
- Achieved substantial numerical sensitivity gains (up to +159.6%) on NumFinE.
- Preserved general semantic performance and demonstrated zero-shot generalization.

Thank you!

Please check our paper and code!

Contact: F09946004@ntu.edu.tw