



## FENCE: Financial Exaggerated Numeral ClassifiEr

---

Sohom Ghosh and Sudip Kumar Naskar

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

May 13, 2023

## ***FENCE: Financial Exaggerated Numeral ClassifiEr***

**Sohom Ghosh<sup>1</sup>, Sudip Kumar Naskar<sup>2</sup>**

<sup>1,2</sup>Jadavpur University, Kolkata, West Bengal, India

<sup>1</sup>sohom1ghosh@gmail.com, <sup>2</sup>sudip.naskar@gmail.com

ORCID: <sup>1</sup>0000-0002-4113-0958, <sup>2</sup>0000-0003-1588-4665

### **Abstract**

The presence of exaggerated numbers i.e. inflated figures in financial texts allude investors to making wrong decisions. This justifies the need for an automated system for detecting such numerals. In this paper, we present **Financial Exaggerated Numeral ClassifiEr (FENCE)** to do so. It consists of a neural network based classifier trained with the context-based embeddings of numerals present in financial market comments and blog titles. FENCE has been deployed in Hugging Face spaces and open sourced under the MIT license.

### **Keywords**

*Exaggerated Numeral Detection, Financial Natural Language Processing, Natural Language Processing*

### **Highlights**

- **Financial Exaggerated Numeral ClassifiEr (FENCE)** is an open-source tool which helps investors to assess if numerals present in financial texts are exaggerated.
- In the back-end it uses context-based domain specific embeddings and neural networks for classification.
- The front-end has been built using Gradio.
- It can be accessed from [GitHub](#) and [Hugging Face Spaces](#).

## Code metadata

C1	Current code version	v1.0.0
C2	Permanent link to code/repository used for this code version	<a href="https://github.com/sohomghosh/FENCE_Financial_Exaggerated_Numeral_Classifier">https://github.com/sohomghosh/FENCE_Financial_Exaggerated_Numeral_Classifier</a>
C3	Permanent link to Reproducible Capsule	<a href="https://codeocean.com/capsule/3497586/tree/v1">https://codeocean.com/capsule/3497586/tree/v1</a>
C4	Legal Code License	MIT
C5	Code versioning system used	git
C6	Software code languages, tools, and services used	Python, Transformers, Gradio, Google Colab
C7	Compilation requirements, operating environments & dependencies	Python packages: numpy, pandas, pickle, torch, transformers, sklearn, gradio
C8	If available Link to developer documentation/manual	<a href="https://github.com/sohomghosh/FENCE_Financial_Exaggerated_Numeral_Classifier/blob/main/README.md">https://github.com/sohomghosh/FENCE_Financial_Exaggerated_Numeral_Classifier/blob/main/README.md</a>
C9	Support email for questions	<a href="mailto:sohom1ghosh@gmail.com">sohom1ghosh@gmail.com</a>

## 1. Introduction

Nowadays, investors tend to read financial contents which are available online while making investment decisions. These contents are generally blog posts, comments about the financial market, etc. Many times, the numerals present in these contents are exaggerated and not correct. We present a tool to address these discrepancies. Firstly, we randomly sample five thousand instances each from market comments and blog titles present in the Numeracy-600K dataset [1]. We extract contextual embedding of the target numeral present in a financial text using SEC-BERT-NUM [2] and pass it through a neural network (NN) based classifier. Due to computational constraints, we train only the NN classifier after freezing the BERT [3] layers. We use 90% of the data for training and on the held-out 10% test set we obtain a weighted F1 score of 0.58. FENCE has been released on the code ocean platform [4] and can be accessed live from Hugging Face spaces<sup>1</sup>.

## 2. Impact Overview

Some of the tools that help users to apply Natural Language Processing in finance include [5], [6], and [7]. Liu et al. [5] developed an online tool, FIN10K for visualizing and analysing various 10K reports. We presented FiNCAT-2 [7] for detecting in-claim and out-of-claim numerals present in financial texts. We further released a comprehensive toolkit FLUEnT [6] which performs various natural language processing tasks on financial texts like readability assessment, hypernym extraction, etc. However, none of the existing tools solve the problem of exaggerated numeral detection in the finance domain. As per our best knowledge, the **FENCE** tool is the first free, open-source and user-friendly tool to do so.

The **FENCE** tool has been released very recently<sup>2</sup>. We expect that this tool will be readily used by investors and researchers to assess whether the numerals present in financial texts are exaggerated or not. For example, the numeral “30%” in the sentence “*Our sales will increase by 30% in the next year*” is not exaggerated. However, if “30%” is changed to “300%” it becomes exaggerated. Automatically detecting such exaggerated numerals helps financial analysts and investors in making decisions and staying away from false claims. Decisions made by expecting false claims to be true often lead to financial losses. **FENCE** uses SEC-BERT-NUM [2] embeddings at the back end. It has been developed using Gradio [8] and hosted on HuggingFace Spaces [9]. It has been open-sourced under the MIT licence. The **FENCE** tool does not require any installation. It can be used from any device with a web browser and an internet connection. It will help in spreading financial knowledge and thereby have a positive impact in improving financial literacy among users. Moreover, while evaluating the

<sup>1</sup>[https://huggingface.co/spaces/sohomghosh/FENCE\\_Financial\\_Exaggerated\\_Numeral\\_Classifier](https://huggingface.co/spaces/sohomghosh/FENCE_Financial_Exaggerated_Numeral_Classifier) (accessed on 8<sup>th</sup> April 2023)

<sup>2</sup>April 2023

performance of **FENCE** on real world datasets consisting of titles of 2,550 news articles, we obtained a weighted average F1 score of 0.63. In future, based on the popularity of the tool, we shall consider releasing a browser based extension with additional capability to process audio files and scan Portable Document Format (PDF) files.

### 3. Back-end and Front-end

As with most other software tools, **FENCE** consists of a back-end and a front-end. As presented in Figure 1, in the back-end it replaces the target numeral present in financial texts with [NUM] token. Subsequently, for this numeral it extracts the contextual embedding of the [NUM] token using SEC-BERT-NUM [2] available on HuggingFace platform [9]. This embedding having 768 dimensions is passed through a neural network having two hidden layers with 512 and 128 neurons respectively. Finally, this network classifies the target numeral as Exaggerated or Non-exaggerated.

We present the user interface of **FENCE** in Figure 2. The inputs have been marked with numbers on the yellow background, while the outputs have been marked with numbers on the orange background. Firstly, in the input text box 1, users are supposed to enter financial texts and press the button marked as 2. The numerals present in the text along with their positions will be highlighted and shown in the panel marked as 3. Tab 4 helps them to choose between if they want to evaluate all or a few specific numerals present in the text. If they decide to evaluate only for specific numerals, they can get the numerals along with their positions in the text using button 5. After this, they can use the multi-select option 6 to choose the specific numerals. On the other hand, for evaluating all the numerals, they can simply click on the “Predict for all numerals” button which appears on selecting the “All numerals” tab (Tab: 4). For brevity, we don’t show this button in Figure 2. Finally, on clicking this button or button 7, the results get presented in panel 8. For the convenience of the users, in the portion marked as 9, we have provided a few sample financial texts.

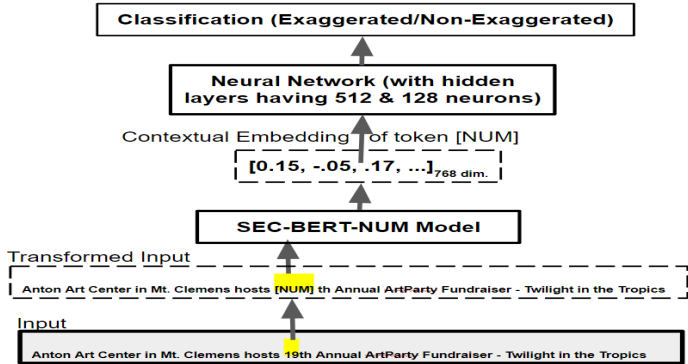


Figure 1: Architecture of **FENCE**

### 4. Limitations and Future Work

As of now, the tool takes only one text as input, and that too in English. In future, we want to enhance its capability so that it seamlessly processes multiple texts in different other languages. Improving the model performance further by experimenting with different other architectures is another direction for future work.

### 5. Declaration of Competing Interest

Financial Exaggerated Numeral ClassifiEr (**FENCE**) has been released for non-commercial academic research. The authors declare that there are no competing and conflicting interests.

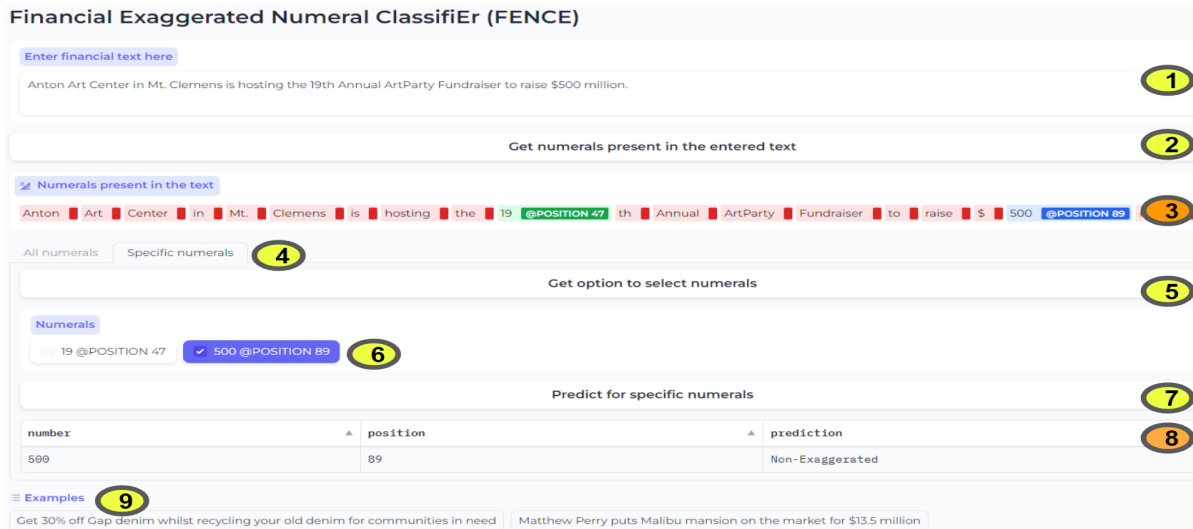


Figure 2: User Interface of FENCE

## References

- [1] C.-C. Chen, H.-H. Huang, H. Takamura, H.-H. Chen, [Numeracy-600K: Learning numeracy for detecting exaggerated information in market comments](#), in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 6307–6313. doi:10.18653/v1/P19-1635. URL <https://aclanthology.org/P19-1635>
- [2] L. Loukas, M. Fergadiotis, I. Chalkidis, E. Spyropoulou, P. Malakasiotis, I. Androutsopoulos, G. Paliouras, [FiNER: Financial numeric entity recognition for XBRL tagging](#), in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 4419–4431. doi:10.18653/v1/2022.acl-long.303. URL <https://aclanthology.org/2022.acl-long.303>
- [3] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.
- [4] S. Ghosh, S. K. Naskar, Fence: Financial exaggerated numeral classifier, <https://www.codeocean.com/> (4 2023). doi:10.24433/C0.1231255.v1.
- [5] Y.-W. Liu, L.-C. Liu, C.-J. Wang, M.-F. Tsai, [Fin10k: A web-based information system for financial report analysis and visualization](#), in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM ’16, Association for Computing Machinery, New York, NY, USA, 2016, p. 2441–2444. doi:10.1145/2983323.2983328. URL <https://doi.org/10.1145/2983323.2983328>
- [6] S. Ghosh, S. K. Naskar, Fluent: Financial language understandability enhancement toolkit, in: 6th Joint International Conference on Data Science & Management of Data (10th ACM IKDD CODS and 28th COMAD) (CODS-COMAD 2023), Association for Computing Machinery, New York, NY, USA, 2023, p. In press. doi:10.1145/3570991.3571067.

- [7] S. Ghosh, S. K. Naskar, [Fincat-2: An enhanced financial numeral claim analysis tool](#), *Software Impacts* 12 (2022) 100288. doi:<https://doi.org/10.1016/j.simpa.2022.100288>.  
URL <https://www.sciencedirect.com/science/article/pii/S2665963822000367>
- [8] A. Abid, A. Abdalla, A. Abid, D. Khan, A. Alfozan, J. Zou, [Gradio: Hassle-free sharing and testing of ml models in the wild](#) (2019). [arXiv:1906.02569](#).
- [9] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, [Transformers: State-of-the-art natural language processing](#), in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Association for Computational Linguistics, Online, 2020, pp. 38–45. doi:[10.18653/v1/2020.emnlp-demos.6](https://doi.org/10.18653/v1/2020.emnlp-demos.6).  
URL <https://aclanthology.org/2020.emnlp-demos.6>

### *Illustrative examples*

**FENCE** has been hosted in Hugging Face Spaces ([https://huggingface.co/spaces/sohomghosh/FENCE\\_Financial\\_Exaggerated\\_Numeral\\_Classifier](https://huggingface.co/spaces/sohomghosh/FENCE_Financial_Exaggerated_Numeral_Classifier)) and can also be accessed from Google Colab (<https://colab.research.google.com/drive/1uLd3DxY1v-xS6e9H512cwyMa5GQXGdHM?usp=sharing>).