MATH FUNCTION RECOGNITION WITH FINE TUNING PRE-TRAINED MODELS

Fatimah Alshamari[1,2] and Abdou Youssef [1]

[1]Department of Computer Science, The George Washington University, Washington D.C, USA
[2]Department of Computer Science, Taibah University, Medina, KSA

ABSTRACT

A Mathematical Function Recognition (MFR) is an important research direction for efficient downstream math tasks such as information retrieval, knowledge extraction, and question answering. The aim of this task is to identify and classify mathematical function into a predefined set of function. However, the lack of annotated data is the bottleneck in the development of an MFR automated model. We begin this paper by describing our approach to creating a labelled dataset for MFR. Then, to identify five categories of mathematical functions, we fine-tuned a set of common pre-trained models: BERT base-cased, BERT baseuncased, DistilBERT-cased, and DistilBERT-uncased. As a result, our contributions in this paper include: (1) an annotated MFR dataset that future researchers can use; and (2) SOTA results obtained by fine tuning pre-trained models for the MFR task. Our experiments demonstrate that the proposed approach achieved a high-quality recognition, with an F1 score of 96.80% on a held-out test set provided by DistilBERT-cased model.

KEYWORDS

Named entity recognition, Math information retrieval, Math language processing, Pre-trained Language models.

1. INTRODUCTION

Mathematics is a highly structured language that typically includes symbols, equations, and expressions with context specific meanings that differs from natural language. In order to extract useful arithmetic information from unstructured content and enable it to be efficiently recognized, it is of considerable significance to develop Math Language Processing (MLP) models for mathematical language. Entity Recognition (ER) is one of the fundamental steps for knowledge acquisition [1], and can help improve the performance of different downstream NLP tasks in mathematics, such as Question Answering (QA), knowledge Extraction (KE), and even augment the effectiveness of search tools and engines, like those in arXiv[1] database [2]. In the domain of mathematics, the ER involves identifying and classifying entities, namely theorems, functions, derivatives of the functions, and other mathematical terms in an input with math content, as shown in Figure 1. Additionally, it can be utilized to draw out associations between entities, such as the relationship between a function and its derivatives. Identifying mathematical terms/expressions is an important task for knowledge acquisition in math language, as it helps to map these terms/expressions to entities that are recognized by natural language models.

[1]Online repository at https://arxiv.org/

David C. Wyld et al. (Eds): DBML, CITE, IOTBC, EDUPAN, NLPAI -2023
pp. 47-56, 2023. IJCI – 2023 DOI:10.5121/ijci.2023.120204

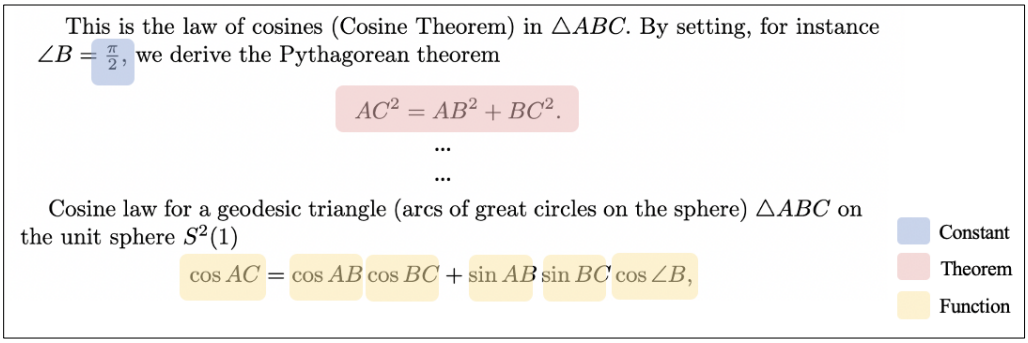

Figure 1. An illustration example of math entities

We hypothesize that this enables better understanding of the mathematical content and consequently leads to a more accurate interpretation. For example, by mapping a query of math expression or formula, such as $a^2 + b^2 = c^2$ to a known entity "*Pythagorean Theorem*", the precision, recall, and overall efficacy of knowledge extraction and search results can be improved. This demonstrates the potential of leveraging math knowledge to enhance the capabilities of search engines. Furthermore, math entity recognition model can be used to construct a knowledge bases, such as domain-specific dictionary, that can be used by downstream tasks [3].

However, developing a universal entity recognition model for mathematical language that is able to accurately recognize all possible entities in the domain is a remarkably challenging task, due to several factors. First, the complexity of the language [4][5], even identifying a single type of entity, such as a function and its corresponding function named entity, is a non-trivial task [2]. In addition, the lack of a varied math entities labeled dataset, and construct such dataset would require a considerable amount of time, efforts, and expertise to develop. Therefore, we believe that breaking down the task into sub-task by focusing on a limited set of entities makes it easier to solve and eliminates the need for large datasets. In addition, by successfully tackling this specific sub-task, it can be demonstrated that the approach used to recognize the chosen entities can be generalized to other types of mathematical entities. To this end, we limited our model to recognize a set of five math function entity.

The main reason we decided to focus on recognizing entities from the functions category is due to the prominent role that functions play in all of mathematics [6]. Furthermore, function entities are quite similar to each other, which makes the separation between them is a demanding task. For instance, the gamma symbol 'Γ' can represent different types of functions, *i.e.,* different entities. Consider for example the following equations:

$$\Gamma(z) = \int_0^\infty e^{-t} t^{z-1}\, dx, \tag{1}$$

$$\Gamma(a, z) = \int_z^\infty t^{a-1} e^{-t}\, dx, \tag{2}$$

$$\Gamma_q(z) = (q;\, q)_\infty (1-q)^{1-z} / (q^z; q)_\infty, \tag{3}$$

$$\Gamma_m(a) = \pi^{\frac{m(m-1)}{4}} \prod_{j=1}^{m} \Gamma\left(a - \frac{1}{2}(j-1)\right). \tag{4}$$

Although every equation includes the gamma symbol 'Γ' indicates the Gamma Function, while equations they each indicate a different type of functions and can relate to different topics. Equation (1) indicates the Gamma Function, while equations (2), (3) and (4) indicate Incomplete-Gamma, q-Gamma and Multivariate-Gamma functions, receptively. For this reason, Ganesalingam in [4] refers to the problem of ambiguity in math language as type-dependent, where a fixed expression can depend on the area of mathematics to which it belongs. In addition, processing math language content required special pre-processing techniques, namely Part-Of-Math tagger (POM) [7], and math function segmentation model [6] for parsing and chunking math expressions and equations.

Besides the problem of ambiguity, function names can have signatures that complicate their recognition. For example, function names could include subscripts, superscripts, and prescripts. This adds another level of challenge with respect to tokenization, where it is hard to determine if the signature is a part of the function name or a separate token [7][5]. Another challenge that has been a stumbling block for many mathematical analytic researchers is the lack of lack of freely-available datasets [8]. Therefore, we had to develop an annotated math function entity recognition dataset. In order to address these limitations, we only focused on **Functions** as the set of mathematical entities. The set includes the following type of functions:

- Exponential Functions.
- Trigonometric Functions.
- Hyperbolic Functions.
- Gamma Functions.
- Bessel Functions:
  - Jν (z), Bessel function of the first kind
  - Yν (z), Bessel function of the second kind

Pre-trained models are able to effectively capture the contextual information of the input, which enables them to better recognize entities in the input [9]. This makes them well-suited for sequence labelling problems such as NER. The successful application pre-trained models to various natural languages that are more complex, such as Chinese [10] and Arabic [11][12], has motivated us to explore their potential for use in mathematical languages as well.

In this study, we propose a Math Function Recognition (MFR) approach based on fine-tuning a set of pre-trained language models to identify mathematical function groups, which is analogous to NER in natural language processing. However, due to the limitations of the lack of labelled datasets and the excessive efforts to create labelled datasets that includes every possible function entity, we restrict the scope of this task to 11 mathematical functions, grouped into five categories, shown in Table 1.

Table 1. The list of functions.

Function's group name	Function name	Example
Gamma	Gamma Function	$P(a+1,z) = P(a,z) - \frac{z^a e^{-z}}{\boldsymbol{\Gamma(a+1)}}$
Exponential	Exponential Function	$P(a+1,z) = P(a,z) - \frac{z^a \boldsymbol{e^{-z}}}{\Gamma(a+1)}$

Bessel	Bessel function of the first kind	$Y_\nu(z) = \frac{\boldsymbol{J_\nu(z)}\ cos\ (\nu\pi)\ -\ \boldsymbol{J_{-\nu}(z)}}{sin\ (\nu\pi)}$
	Bessel function of the second kind	$\boldsymbol{Y_\nu(z)} = \frac{J_\nu(z)\ \cos\ (\nu\pi)\ -\ J_{-\nu}(z)}{sin\ (\nu\pi)}$
Trigonometric	Cosine Function	$Y_\nu(z) = \frac{J_\nu(z)\ \boldsymbol{cos\ (\nu\pi)}\ -\ J_{-\nu}(z)}{sin\ (\nu\pi)}$
	Sine Function	$Y_\nu(z) = \frac{J_\nu(z)\ cos\ (\nu\pi)\ -\ J_{-\nu}(z)}{\boldsymbol{sin\ (\nu\pi)}}$
	Tangent Function	$\boldsymbol{tan(2z)} = \frac{2\ \boldsymbol{tan\ z}}{1 - \boldsymbol{tan^2 z}} = \frac{2\ cos\ z}{cos^2 z - 1} = \frac{2}{cot\ z\ -\ \boldsymbol{tan\ z}}$
	Cotangent Function	$\frac{\partial}{\partial \nu} P_\nu^\mu(x) = \pi\boldsymbol{cot\ (\nu\pi)} P_\nu^\mu(x) - \frac{1}{\pi} A_\nu^\mu(x)$
	Cosecant Function	$J_\nu(z) = \boldsymbol{csc(\nu\pi)}\ (\ Y_{-\nu}(z) -\ Y_\nu(z)\ cos\ (\nu\pi))$
Hyperbolic Trigonometric	Hyperbolic cosine function	$\boldsymbol{cosh\ z} = 1 + \frac{z^2}{2!} + \frac{z^4}{4!} + \ldots$
	Hyperbolic sine function	$bei_n(x\sqrt{2}) = \frac{(-1)^n}{\pi} \int_0^\pi sin\ (x\ sin\ t - nt)\ \boldsymbol{sinh\ (x\ sin\ t)} dt$

2. RELATED WORK

Name Entity Recognition (NER) is one of the fundamental NLP tasks that identify and categorize entities in text based on their roles, such as people, organizations, and locations. Recently, fine-tuning contextualized pre-trained language models such as BERT [13], RoBERTa [14], or DistilBERT [15] have yielded promising results in the NER task on a variety of datasets, languages, and domains [13], [16], [17]. This section provides an overview of some related NER works using the methodology of pre-trained language models. Studies have shown that fine-tuning contextualized models can improve the performance of NER models on various datasets and domains.

Strubell *et al.* in [18] showed that DistilBERT outperformed BERT on a newswire NER dataset, while Jiao *et al.* in [19] showed that DistilBERT achieved comparable performance to BERT on more general datatset, which consists of news, broadcast, and web text.

Recently, NER models have been widely used and applied in the scientific domains that have entities that can be identified and extracted. In medical domain, the contextualized models, including BERT-base and DistilBERT, has been used for pre-training and fine-tuning over medical data to extract biomedical entities [20] In addition, several studies have shown that fine-tuning BERT can improve the performance of NER models on medical datasets in different languages, [21] for Arabic, [22] for Chinese, and [23] for Spanish.

Similarly, there have been studies that have fine-tuned pre-trained language models, including BEET and DistilBERT, on chemistry datasets for recognizing entities such as chemical compounds or for chemical relation extraction (RE) [24].

To the best of our knowledge, there are still no fine-tuned and assessed the performance of BERT or DistilBERT models on math entity recognition problem. In this case, we take our initial step to explore existing models and fine-tune them on our development training and developing dataset.

3. DATASET

For the purposes of evaluating a given task in the Math Language Processing domain a math specific data is needed. Hence, we utilized the Simple-XML dataset[2] [25], that have been derived from the Digital Library of Mathematical Functions (DLMF) from the National Institute of Standards and Technology (NIST)[3] [26]. This dataset consists of: 20,040 sentences; 25,930 math elements; and 8,494 numbered equations in a semi-structured form.

Although this dataset was obtained from the specified math domain, it was not labelled to carry out a specific supervised MLP task. As a result, for the MFR task, we tagged the function entities in the input text with their associated type. The Simple-XML dataset has more than 250 distinct types of functions, however because labeling is tedious and time consuming, we only include the equation subset, which consists of 8,488 sentences with about 10,000 instances of functions. It is important to note that a math statement can be composed of an equation with one or more functions.

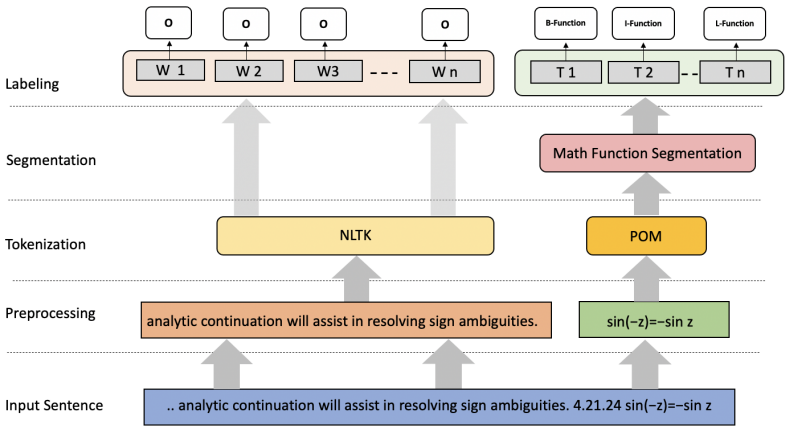

Figure 2. Labeling the dataset

3.1. Dataset Setup

To be able to carry this task, a labelled dataset is needed as training data, we follow the processing pipeline that is illustrated in Figure 2. Specifically, to reduce the efforts needed to annotate a math

2 https://github.com/abdouyoussef/math-dlmf-dataset

3 DLMF at https://dlmf.nist.gov/

function entity, we follow a three phases methodology: a tokenization phase, a segmentation phase, followed by labeling phase.

For the initial phase, we tokenize the input sentence by applying the English tokenizer in Natural Language Toolkit (NLTK) [27] over the text, and applying the Part-of-Math (POM) tagger [7] over the math content. Given the tokenized text and math content, the task is to recognize the function in a given math expression.

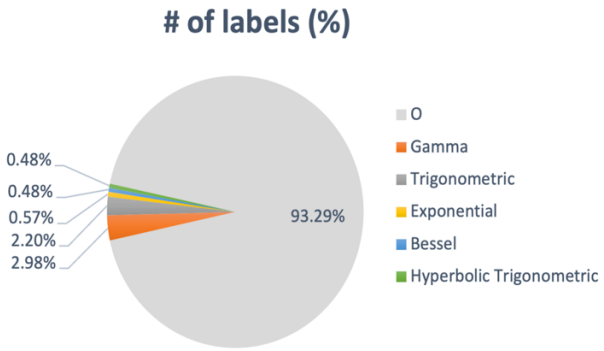

Figure 3. Statistics of different types of function entities

For the second phase, we apply the math function segmentation model [6] which recognize a selection of functions in a tokenized input that are written in the following formats:

- $f(x)$
- $f(x,y)$
- $f_n(x)$
- $f'(y)$
- $f_m^+(x)$
- ${}_p f_q(a_1,\ldots,a_p;b_1,\ldots,b_q)$

The segmentation model detects the function boundaries, thus simplify the labeling process. For the labeling phase we rely on regular-expression patterns that capture a set of function as illustrated in Table 1, and we followed the BILO (Begin, Inside, Last, Outside) schema [28].

Specifically, in our work, all text tokens are labeled with an 'O' (outside) label, while the math tokens are labeled 'O' only if it is determined to be outside of any function boundaries. In case a math token is identified as a part of a function entity, it is labeled based on its position in the function. The first token of each recognized function must be labled with *'B-Entity-Name'* tag, while the last token is labeled with *'L-Entity-Name'*. The sequence of tokens between the both (Beginning and Last) labels are labeled with the tag '*I-Entity-Name*'. It is important to noted that all three tags must share the same entity name. The distribution of the entities in the final annotated datset is shown in Figure 3.

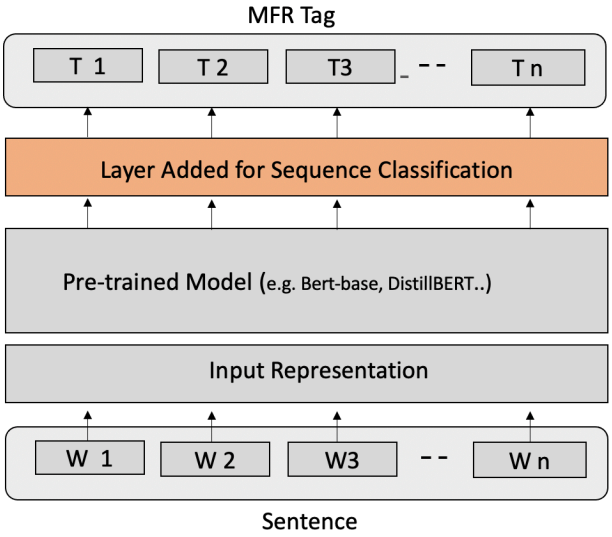

Figure 4. Fine-tuning pre-trained model for MFR task

5. EXPERIMENTAL RESULTS

Recently, tuning pre-trained language models such as BERT, RoBERTa, XLNet, etc preformed well for many NLP tasks including NER [13]. Hence, we hypothesize that fine-tinging a pre-trained language model will be able to perform for MFR task.

We approach the MFR task in the same manner as the NER task, which is a sequence labeling task in which each word is assigned a tag. The input sequence is fed into a pre-trained contextualized language model, which generates context embeddings for each word. To recognize the tags, these embeddings are tuned with a task-specific linear feed forward layer at the output layer. The architecture of fine-tinging a pre-trained model for MFR task is shown in Figure 4.

We used transformer pre-trained language models from the Hugging face library [29] to fine-tune four pre-trained language models: BERT base-cased, BERT base-uncased, DistilBERT-cased, and DistilBERT-uncased [13] [15]. For fine-tuning, we used the same hyperparameters setup for all models, including the Adam optimizer with a learning rate of 3e-5, warmup proportion of 0.1, gradient clipping of 1.0, and weight decay of 0.01. Additionally, we set a maximum sequence length of 500, number of epochs of 4, and a batch size of 16. The data were randomly sampled into 60% for training and 20% each for validation and testing. We then evaluated the performance of the models on the test set primarily, and report the results using F1-score, precision, and recall scores.

Table 2. Performance appraisals based on accuracy, recall, and F1- Pre-trained degrees of fine tuning for different models test data

Model	**Precision**	**Recall**	**F1-score**
BERT-base-cased	98.46	95.07	96.72
BERT-base-uncased	95.27	88.8	91.84
DistilBERT-cased	99.31	94.64	**96.80**
DistilBERT-uncased	98.61	94.23	96.34

We conduct ablation study of the models' performance for each class of functions as shown in Table 3. It can be seen that the Hyperbolic Trigonometric function class has the higher overall results. The Trigonometric and Gamma entities have a good balance of precision and recall. The model is performing better in predicting the Exponential entity than the Bessel, but not as well as the Gamma or Trigonometric. Bessel has the lowest score across the models, but is still quite high. It is likely that this difference is due to the fact that there is less annotated data for this entity type. Overall, the results of the set of fine-tuned BERT models are quite successful.

Table 3. Performance evaluation of each type of function based on DistilBERT-cased model

Entity	**Precision**	**Recall**	**F1-score**	**Support**
Bessel	97.22	83.33	89.74	42
Gamma	99.15	97.48	98.31	119
Hyperbolic Trigonometric	**100**	**100**	**100**	79
Trigonometric	99.02	94.39	96.65	321
Exponential	100	90.91	95.24	55

6. CONCLUSION

This study presents the fine-tuned of pre-trained language models for math function NER task. In addition to producing an annotated FNER dataset. The proposed models achieved significant performance based on F1-Score. However, all fine-tuned models that are used in this study were pre-trained on text corpora not on math corpora. Therefore, it would be interesting to additionally investigate pre-training model from scratch on math corpus to develop a domain-specific; a pre-trained math language model.

REFERENCES

[1] Moll´a, D., Van Zaanen, M., and Smith, D. (2006). Named entity recognition for question answering. In Proceedings of the Australasian language technology workshop 2006, pages 51–58.

[2] Greiner-Petter, A., Youssef, A., Ruas, T., Miller, B. R., Schubotz, M., Aizawa, A., and Gipp, B. (2020). Math-word embedding in math search and semantic extraction. *Scientometrics*, 125:3017–3046.

[3] Shang, J., Liu, L., Ren, X., Gu, X., Ren, T., and Han, J. (2018). Learning named entity tagger using domain-specific dictionary. *arXiv preprint arXiv*:1809.03599.

[4] Ganesalingam, M. (2013). The language of mathematics. In *The Language of Mathematics*, page17–38. Springer.

[5] Priss, U. (2019). Applying semiotic-conceptual analysis to mathematical language. In *Graph Based Representation and Reasoning: 24th International Conference on Conceptual Structures, ICCS 2019*, Marburg, Germany, July 1–4, 2019, Proceedings 24, pages 248–256. Springer.

[6] Alshamari, F.,Youssef, A. (in press). Math Chunking and Function Recognition using Deep Learning. To appear in *Proceedings of the 21st IEEE International Conference On Machine Learning And Applications (ICMLA)*

[7] Youssef, A. (2017). Part-of-math tagging and applications. In *The International Conference on Intelligent Computer Mathematics*, pages 356–374. Springer.

[8] Meadows, J. and Freitas, A. (2022). A survey in mathematical language processing. *arXiv preprint arXiv:2205.15231*.

[9] Hadiwinoto, C., Ng, H. T., and Gan, W. C. (2019). Improved word sense disambiguation using pre-trained contextualized word representations. *arXiv preprint arXiv:1910.00194*.

[10] Yu, P. and Wang, X. (2020). Bert-based named entity recognition in chinese twenty-four histories. In *Web Information Systems and Applications: 17th International Conference, WISA* 2020, Guangzhou, China, September 23–25, 2020, Proceedings 17, pages 289–301. Springer.

[11] Alsaaran, N. and Alrabiah, M. (2021). Classical arabic named entity recognition using variant deep neural network architectures and bert. *IEEE* Access, 9:91537–91547

[12] Al-Qurishi, M. S. and Souissi, R. (2021). Arabic named entity recognition using transformer-based-crf model. In *Proceedings of The Fourth International Conference on Natural Language and Speech Processing (ICNLSP 2021*), pages 262–271.

[13] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[14] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

[15] Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.*arXiv preprint arXiv:1910.01108*.

[16] Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

[17] Jiang, S., Zhao, S., Hou, K., Liu, Y., Zhang, L., et al. (2019). A bert-bilstm-crf model for chinese electronic medical records named entity recognition. In *2019 12th international conference on intelligent computation technology and automation (ICICTA)*, pages 166– 169. IEEE.

[18] Strubell, E., Ganesh, A., and McCallum, A. (2019). Energy and policy considerations for deep learning in nlp. *arXiv preprint arXiv:1906.02243*.

[19] Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., and Liu, Q. (2019). Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*.

[20] Akhtyamova, L. (2020). Named entity recognition in spanish biomedical literature: short review and bert model. In *2020 26th Conference of Open Innovations Association (FRUCT)*, pages 1–7. IEEE.

[21] Boudjellal, N., Zhang, H., Khan, A., Ahmad, A., Naseem, R., Shang, J., and Dai, L. (2021). Abioner: a bert-based model for arabic biomedical named-entity recognition. *Complexity*, 2021:1–6.

[22] Dai, Z., Wang, X., Ni, P., Li, Y., Li, G., and Bai, X. (2019). Named entity recognition using bert bilstm crf for chinese electronic health records. In *2019 12th international congress on image and signal processing, biomedical engineering and informatics (cisp-bmei)*, pages 1– 5. IEEE.

[23] Tamayo, A., Burgos, D. A., and Gelbukh, A. (2022). mbert and simple post-processing: A baseline for disease mention detection in spanish. In *Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings*.

[24] Isazawa, T. and Cole, J. M. (2022). Single model for organic and inorganic chemical named entity recognition in ChemDataExtractor. *Journal of Chemical Information and Modeling*, 62(5):1207–1213.

[25] Youssef, A. and Miller, B. R. (2020). A contextual and labeled math-dataset derived from nist's dlmf. *In Intelligent Computer Mathematics: 13th International Conference, CICM 2020*, Bertinoro, Italy, July 26–31, 2020, Proceedings 13, pages 324–330. Springer.

[26] Olver, F., Daalhuis, A., Lozier, B., Schneider, B., Boisvert, R., Clark, C., Miller, B., and Saunders, B. (2016). Nist digital library of mathematical functions http://dlmf. nist. gov. *Release,* 1:22.

[27] Bird, S. (2006). NLTK: The Natural Language Toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions,* pages 69–72.

[28] Carpenter, B. (2009). Coding chunkers as taggers: Io, bio, bmewo, and bmewo+. *LingPipe Blog*, page 14.

[29] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

AUTHORS

Fatimah Alshamari is a doctoral candidate in the Department of Computer Science at the George Washington University. She received B.C.Sc (Bachelor of Computer Science) degree in 2007,and M.C.Sc (Master of Computer Science) degree in 2014. She is now Assistant Lecturer of Taibah University. Her research interests include Data Mining, Mathematics Language Processing**,** Knowledge Extraction.

Abdou Youssef has 30 years of research and teaching experience in the field of computer science. He is currently a tenured Professor at The George Washington University, Washington, D.C, which he joined as Assistant Professor in Fall of 1987. His current research interests are applied data science, math search and math language processing, audio-visual data processing, pattern recognition, theory and algorithms. He has published over 125 papers in those areas, and co-edited the book Interconnection Networks for High-Performance Parallel Computer Published by IEEE Computer Society Press in 1994. His research has been funded by NSF, NSA, and NIST. Currently, he is developing novel techniques for part-of-math tagging, math semantics extraction and question answering, and big-data applications such as fraud detection in the retail business, next-generation recommendation systems, and more.