Addax at WojoodNER 2024: Attention-Based Dual-Channel Neural Network for Arabic Named Entity Recognition

Issam Ait Yahia[1]*, **Houdaifa Atou**[1]*, **Ismail Berrada**[1],
[1]College of Computing
Mohammed VI Polytechnic University, Ben Guerir, Morocco
{issam.aityahia, houdaifa.atou, ismail.berrada}@um6p.ma

Abstract

Named Entity Recognition (NER) is a fundamental task in Natural Language Processing that focuses on extracting entities such as names of people, organizations, locations, and dates from text. Despite significant advancements due to deep learning and transformer architectures, NER still faces challenges, particularly in low-resource languages like Arabic. This paper presents a BERT-based NER system that utilizes a two-channel parallel hybrid neural network with an attention mechanism specifically designed for the NER Shared Task 2024. For the WojoodNER 2024 Shared Task, our approach ranked second by scoring 90.13% in micro-F1 on the test set. The results demonstrate the effectiveness of combining advanced neural network architectures with contextualized word embeddings in improving NER performance for Arabic. Code is available at https://github.com/issam-yahya/Addax.

1 Introduction

Named Entity Recognition (NER) is one of the main tasks of Natural Language Processing (NLP) and plays a fundamental role in different NLP applications, such as machine translation, information retrieval, and question-answering (Alami et al., 2023). NER involves extracting entities from unstructured text, including people's names, organizations' names, locations, and dates. These entities can be categorized as flat or nested based on their structure. Flat entities consist of a single non-overlapping span of text representing a single distinct entity. Nested entities, on the other hand, include entities that are embedded within other entities.

In this paper, we present our participating system for the second edition of the flat subtask of NER Shared Task 2024 (Jarrar et al., 2024), building on the foundations laid by the first edition (Jarrar et al., 2023). Our system is a BERT-based model that utilizes a two-channel parallel hybrid neural network with an attention mechanism. The best results were achieved using AraBERT (Antoun et al., 2020). We evaluated our system's performance using the micro-average F1 score, Recall, and Precision. On the test set, it achieved a micro-F1 score of 90.13% and secured the second position.

2 Related Work

Previous studies in Named Entity Recognition (NER) have employed diverse methodologies, including knowledge-based approaches, machine learning techniques, and deep learning methods. Knowledge-based approaches were among the earliest strategies used in NER. These methods relied on hand-crafted rules (Hanisch et al., 2005; Shaalan and Raza, 2007) and on language-specific knowledge such as lexical markers (Zhang and Elhadad, 2013) and entity dictionaries (Etzioni et al., 2005). With the rise of machine learning, studies shifted from hand-crafted rules to various machine learning techniques such as Support Vector Machines (SVM) (McNamee and Mayfield, 2002), Hidden Markov Models (HMM) (Morwal et al., 2012), and Conditional Random Fields (CRF) (McCallum and Li, 2003). Recently, significant progress in NER has been driven by advancements in deep learning (Li et al., 2022). Various deep learning architectures have been successfully incorporated into NER, including Convolutional Neural Networks (CNNs) (Gui et al., 2019), Recurrent Neural Networks (RNNs) (Lample et al., 2016), and Transformers (Labusch et al., 2019; Yan et al., 2019).

Recent models leveraging transformer architectures, such as BERT (Devlin et al., 2018) and its variants, have set new benchmarks in NER performance by capturing more contextual nuances and complex language structures than traditional models. Additionally, improvements in pretrained

*These authors contributed equally to this work.

Proceedings of The Second Arabic Natural Language Processing Conference, pages 867–873

language models (Devlin et al., 2018; Antoun et al., 2020) have enabled the adaptation of NER systems developed with fewer annotated data to new languages and domains, improving their accuracy and performance. However, NER still faces many challenges, especially in low-resource languages such as Arabic (Darwish et al., 2021).

Most Arabic NER research has focused on flat entities, primarily targeting a few coarse-grained entity types, such as person, organization, and location. Although coarse-grained NER is useful across various domains and serves as a foundational approach, it falls short for tasks that demand a more detailed understanding of named entities (Ling and Weld, 2012). In this context, $Wojood_{Fine}$ (Liqreina et al., 2023) was introduced to address the scarcity of fine-grained Arabic NER corpora. It is an extension of Wojood (Jarrar et al., 2022) that introduced finer-grained sub-types. Wojood initially included 21 entity types and was primarily gathered from Modern Standard Arabic (MSA) articles, which constituted the majority of the data, with a smaller portion collected from social media in Palestinian and Lebanese dialects. $Wojood_{Fine}$ enhanced the original Wojood corpus by introducing fine-grained entities to four entity types: Geopolitical Entity (GPE), Organization (ORG), Location (LOC), and Facility (FAC).

3 Data

The Wojood Fine corpus extends annotation of the Wojood original corpus (Jarrar et al., 2022) by including fine-grained annotations for named-entity subtypes. It contains 550K tokens and was manually annotated with 21 entity types. It is worth mentioning that approximately 80% of Wojood originates from Modern Standard Arabic (MSA) articles, but about 10% comes from content on social media sites that were written in Palestinian or Lebanese dialects via the Curras and Baladi corpora (Al-Haff et al., 2022). Additionally, Wojood has also nested named entities, though some of them could refer to more than one type of entity like 'Organization', which may interfere with information retrieval tasks further downstream. In response, $Wojood_{Fine}$ provides subtypes for four main categories of entities: GPE, ORG, LOC, and FAC.

3.1 Main Entity Types

Table 1 provides an overview of the frequency of these four main entity types in both Wojood and $Wojood_{Fine}$.

Tag	Wojood	$Wojood_{Fine}$
GPE	21,780	23,085
ORG	18,785	18,747
LOC	917	1,441
FAC	1,215	1,121
Total	**42,697**	**44,394**

Table 1: Frequency of the four entity types in Wojood and $Wojood_{Fine}$.

The development of $Wojood_{Fine}$ was informed by LDC's ACE 2008 annotation guidelines for Arabic entities v7.4.2.

3.2 Sub-types

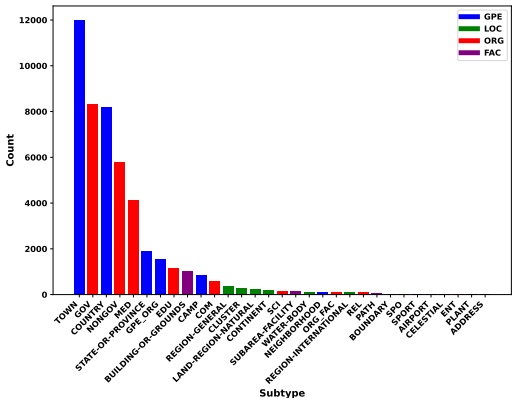

Figure 1: Distribution of Entity Subtypes in $Wojood_{Fine}$ Corpus.

The sub-types introduced in $Wojood_{Fine}$ provide a more granular level of annotation for GPE, ORG, LOC, and FAC entities, resulting in the addition of 31 new entity sub-types. The frequency of each subtype in the $Wojood_{Fine}$ corpus is shown in Figure 1.

$Wojood_{Fine}$ is available as a RESTful web service, and both the data and source code are publicly accessible [1]

4 System Overview

4.1 Data Pre-processing

Existing NER models use nested tagging formats like BIO, assigning tags to both primary entity types and their sub-types (El Mekki et al., 2022)

[1] https://github.com/SinaLab/ArabicNER

(e.g., B-GPE, I-GPE for Geopolitical Entities and B-CAMP, B-CLUSTER for sub-types). While informative, this method is complex due to the need to predict many independent tags. To address this challenge, we used combined tagging to classify entities. In this approach, we combined the main entity type and its sub-types into a single category. For example, the entity "Palestine" with a main type of B-GPE and a sub-type of B-COUNTRY, will be classified as "B-GPE+B-COUNTRY".

Combined tags follow the BIO scheme for entity boundaries, with "I-" and "B-" indicating the inside and beginning of entities, respectively. This approach simplifies training by focusing on a single combined tag per entity, integrating both main and sub-type information. It reduces complexity despite a slight reduction in granularity and potential challenges with complex nested entities.

4.2 Model Architecture

4.2.1 Architecture Overview

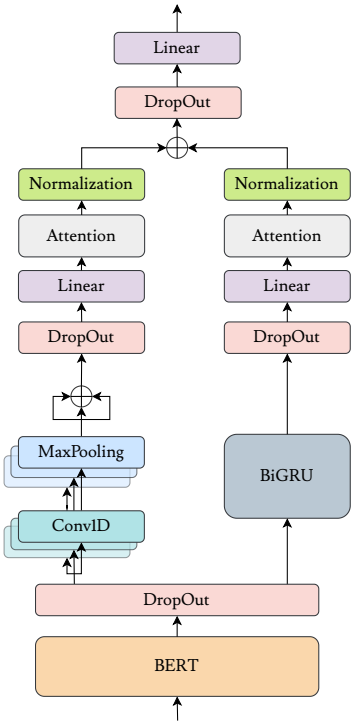

Figure 2: Overall Model Architecture.

Our model, as shown in Figure 2, utilizes a two-channel parallel hybrid neural network with an attention mechanism (Chen et al., 2023) to address the unique characteristics of the Arabic language. Each Arabic word possesses both local properties and global properties, necessitating a specialized approach for effective processing. To achieve this, our model uses BERT embeddings (Devlin et al., 2018) to generate contextualized word representations and utilizes two channels, each with a specific purpose. The first channel focuses on local feature extraction using CNNs, while the second channel captures long-range dependencies using RNNs, enabling the model to understand complex linguistic concepts within sentences.

4.2.2 Convolutional Layers Channel

The first channel of our model uses 3 stacked Conv1D layers to extract local patterns and features from the embeddings. Max pooling is applied to reduce feature map dimensions, allowing convolutional filters to focus on significant details and further preventing overfitting. Dropout regularization is incorporated to prevent overfitting by randomly nullifying some input units during training, thereby enhancing generalization. An ablation study was conducted, as shown in Table 4, comparing configurations with 1, 3, and 4 convolutional heads to assess the impact of the number of heads on performance.

4.2.3 Bidirectional GRU Channel

The second channel uses Bidirectional GRU layers to account for long-range dependencies in the input sequences, enhancing the model's ability to recognize entities. This channel captures the sequence context bidirectionally, allowing the model to understand the context of each word within a sentence more effectively. We performed an analysis comparing the use of Bidirectional GRU with Bidirectional LSTM layers to evaluate their effectiveness, detailed in Table 2.

4.2.4 Attention Mechanism

The attention mechanism (Vaswani et al., 2017) plays a role in our architecture by weighing different parts of the input sequence, concentrating on the most relevant features for prediction. This mechanism significantly enhances the model's performance by allowing it to focus on the most critical parts of the input data.

Moreover, input normalization stabilizes the training process, and additional dropout layers ensure robust classification without overfitting. The integration of the attention mechanism consistently improved results across various configurations, as shown in Table 2.

Model	F1	R	P
AraBERTv0.2 + linear	86.74	88.33	85.21
AraBERTv0.2-Twitter + linear	89.16	88.65	89.67
AraBERTv0.2-Twitter + BiGRU + Attention	88.20	88.20	88.20
AraBERTv0.2-Twitter + Conv1D + Attention	88.90	88.39	88.91
AraBERTv0.2-Twitter + Conv1D + BiGRU	89.11	88.59	89.97
AraBERTv0.2-Twitter + Conv1D + BiLSTM + Attention	89.30	89.33	89.28
AraBERTv0.2-Twitter + Conv1D + BiGRU + Attention	**89.32**	**88.67**	**89.97**

Table 2: The results of our model on Subtask 1. The best results are highlighted in bold. F1: micro-F1 score, R: Recall, P: Precision.

Team Name	Micro-F1
notfine_tuning	91
muNERa	90
Addax *(Ours)*	**90**
Baseline	89
DRU - Arab Center	85
Bangor	86

Table 3: Official leaderboard of Subtask-1

#Conv	F1	R	P
1	86.74	88.33	85.21
3	**89.16**	**88.65**	**89.67**
4	88.73	89.39	88.07

Table 4: Ablation Study: Number of convolutional blocks.

4.2.5 Attention-Based Dual-Channel Neural Network Algorithm

Algorithm 1 presents the detailed process flow of the system. The input sequences are initially processed using BERT, followed by a dropout layer to reduce overfitting. In the local feature extraction channel, the embeddings' dimension is permuted and passed to stacked convolutional layers. After convolution, the output is permuted back to its original dimension, passed through another dropout layer, and then linearly projected. In parallel, global features are extracted using a Bidirectional GRU, followed by a dropout layer and a linear projection. The outputs from local and global feature extraction channels are then independently processed using an attention mechanism and subsequently normalized. The outputs of both channels are then concatenated, and a final dropout and linear projection layer are applied to the resulting features to predict entity labels.

Entity	Precision	Recall	F1-Score
CARDINAL	87.35	85.29	86.31
CURR	0.00	0.00	0.00
DATE	92.98	94.03	93.50
EVENT	69.93	73.40	71.63
FAC	72.45	82.56	77.17
GPE	91.87	95.16	93.49
LANGUAGE	70.59	75.00	72.73
LAW	80.39	87.23	83.67
LOC	83.33	87.96	85.59
MONEY	56.25	81.82	66.67
NORP	72.18	73.03	72.60
OCC	85.82	88.33	87.06
ORDINAL	94.12	93.48	93.80
ORG	90.28	93.62	91.92
PERCENT	100.00	100.00	100.00
PERS	94.15	94.45	94.30
PRODUCT	57.14	50.00	53.33
QUANTITY	50.00	66.67	57.14
TIME	72.22	78.79	75.36
UNIT	0.00	0.00	0.00
WEBSITE	64.94	62.50	63.69

Table 5: Precision, Recall, and F1-Scores by Entity Types

5 Experiment and Results

Our model was implemented using PyTorch (Paszke et al., 2019), and Transformers from HuggingFace (Wolf et al., 2019). The model was trained for 10 epochs on an NVIDIA RTX-A6000 GPU, with a batch size of 32 and a learning rate of 1×10^{-3}. We used the officially provided training, validation, and test splits to train, evaluate, and test our model's performance. We have evaluated the performance of our model based on precision, recall, and micro-F1 score, as mentioned in the shared task guideline. The experimental results are summarized in Table 2.

Algorithm 1: Attention-Based Dual-Channel Neural Network

Input: Sequence IDs, mask
Output: Predicted entity labels
$h_e \leftarrow \text{BERT}(x)$;
$h_d \leftarrow \text{Dropout}(h_e)$;
Local Feature Extraction
$h_p \leftarrow \text{permute}(h_d, 2, 1)$;
$h_c \leftarrow \text{StackedConv1D}(h_p)$;
$h_p \leftarrow \text{permute}(h_c, 2, 1)$;
$h_p \leftarrow \text{Dropout}(h_p)$;
$c_l \leftarrow \text{Linear}(h_p, d_c)$;
Global Feature Extraction
$h_b \leftarrow \text{BiGRU}(h_d)$;
$h_b \leftarrow \text{Dropout}(h_b)$;
$h_g \leftarrow \text{Linear}(h_b, d_g)$;
Attention
$a_1 \leftarrow \text{MHAttn}(h_g, h_g, h_g)$;
$a_1 \leftarrow \text{LayerNorm}(a_1)$;
$a_2 \leftarrow \text{MHAttn}(c_l, c_l, c_l)$;
$a_2 \leftarrow \text{LayerNorm}(a_2)$;
Combine & Predict
$f \leftarrow \text{concat}([a_1, a_2], dim = 2)$;
$f_d \leftarrow \text{Dropout}(f)$;
$\hat{y} \leftarrow \text{Linear}(f_d, C)$;

Our experiments showed that the base AraBERTv0.2-Twitter (Antoun et al., 2020) outperformed AraBERTv0.2. Additionally, the best-performing model was the combination of AraBERTv0.2-Twitter with Conv1D, BiGRU, and Attention, which achieved an F1 score of 89.32% on the evaluation set, highlighting the benefits of combining these techniques for this specific task.

Table 5 displays the model's performance across different classes. High-performing entities, including PERCENT, PERS, ORDINAL, DATE, and GPE, exhibit F1 scores exceeding 93%. In contrast, entities such as PRODUCT, CURR, and UNIT have significantly lower F1 scores, underscoring the need for further improvements.

As shown in Table 3, the first place in Subtask-1 was secured by **notfine_tuning**, achieving a micro-F1 score of 91%. Our team, **Addax**, closely followed with a micro-F1 score of 90%, tying for second place.

6 Discussion

Our experimental results demonstrate that integrating a BERT-based model with CNNs, BiGRU, and an attention mechanism in a two-channel network enhances the base model's performance. The ablation studies confirm that each component contributes to improved outcomes, validating the robustness of our architecture. Moreover, the AraBERTv0.2-Twitter model outperforms the AraBERTv0.2 model, suggesting that pretraining on Twitter data more effectively captures the structural nuances of social media text, which aligns better with the WojoodNER 2024 corpus. Future research should explore multi-head models and address low-represented classes to further enhance performance. Focusing on these areas could provide deeper insights and improvements in Arabic Named Entity Recognition (NER).

7 Conclusion

In this paper, we introduced the architecture we used for the NER Shared Task 2024, Subtask-1. Our model is based on a BERT language model and uses a two-channel parallel neural network with an attention mechanism. Our architecture combines a Convolutional Layers Channel and a Bidirectional GRU Channel to extract local and long-range dependencies within sentences. We found that the attention mechanism further improves the model's performance by allowing the model to focus on relevant parts of the input sequence. As shown in our ablation studies, all components of the model contributed to enhancing the final results. Our model achieved a micro-F1 score of 90.13% on the test set for Subtask-1 of WojoodNER 2024 Shared Task, securing the second position.

References

Karim Al-Haff, Mustafa Jarrar, Tymaa Hammouda, and Fadi Zaraket. 2022. Curras + baladi: Towards a Levantine corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 769–778, Marseille, France. European Language Resources Association.

Hamza Alami, Abdelkader El Mahdaouy, Abdessamad Benlahbib, Noureddine En-Nahnahi, Ismail Berrada, and Said El Alaoui Ouatik. 2023. Daqas: Deep arabic question answering system based on duplicate question detection and machine reading comprehension. *Journal of King Saud University - Computer and Information Sciences*, 35(8):101709.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. In *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, page 9.

Na Chen, Yanqiu Sun, and Yan Yan. 2023. Sentiment analysis and research based on two-channel parallel hybrid neural network model with attention mechanism. *IET Control Theory & Applications*, 17.

Kareem Darwish, Nizar Habash, Mourad Abbas, Hend Al-Khalifa, Huseein T Al-Natsheh, Houda Bouamor, Karim Bouzoubaa, Violetta Cavalli-Sforza, Samhaa R El-Beltagy, Wassim El-Hajj, et al. 2021. A panoramic survey of natural language processing in the arab world. *Communications of the ACM*, 64(4):72–81.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Abdellah El Mekki, Abdelkader El Mahdaouy, Ismail Berrada, and Ahmed Khoumsi. 2022. Adasl: An unsupervised domain adaptation framework for arabic multi-dialectal sequence labeling. *Information Processing & Management*, 59(4):102964.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artificial intelligence*, 165(1):91–134.

Tao Gui, Ruotian Ma, Qi Zhang, Lujun Zhao, Yu-Gang Jiang, and Xuanjing Huang. 2019. Cnn-based chinese ner with lexicon rethinking. In *ijcai*, volume 2019.

Daniel Hanisch, Katrin Fundel, Heinz-Theodor Mevissen, Ralf Zimmer, and Juliane Fluck. 2005. Prominer: rule-based protein and gene entity recognition. *BMC bioinformatics*, 6:1–9.

Mustafa Jarrar, Muhammad Abdul-Mageed, Mohammed Khalilia, Bashar Talafha, AbdelRahim Elmadany, Nagham Hamad, and Alaa' Omar. 2023. Wojoodner 2023: The First Arabic Named Entity Recognition Shared Task. In *Proceedings of the 1st Arabic Natural Language Processing Conference (ArabicNLP), Part of the EMNLP 2023*, pages 748–758. ACL.

Mustafa Jarrar, Nagham Hamad, Mohammed Khalilia, Bashar Talafha, and Muhammad Elmadany, Abdel-Rahim Abdul-Mageed. 2024. WojoodNER 2024: The Second Arabic Named Entity Recognition Shared Task. In *Proceedings of the 2nd Arabic Natural Language Processing Conference (Arabic-NLP), Part of the ACL 2024*. Association for Computational Linguistics.

Mustafa Jarrar, Mohammed Khalilia, and Sana Ghanem. 2022. Wojood: Nested arabic named entity corpus and recognition using bert. *arXiv preprint arXiv:2205.09651*.

Kai Labusch, Preußischer Kulturbesitz, Clemens Neudecker, and David Zellhöfer. 2019. Bert for named entity recognition in contemporary and historical german. In *Proceedings of the 15th conference on natural language processing, Erlangen, Germany*, pages 8–11.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.

Xiao Ling and Daniel Weld. 2012. Fine-grained entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 94–100.

Haneen Liqreina, Mustafa Jarrar, Mohammed Khalilia, Ahmed Oumar El-Shangiti, and Muhammad Abdul-Mageed. 2023. Arabic Fine-Grained Entity Recognition. In *Proceedings of the 1st Arabic Natural Language Processing Conference (ArabicNLP), Part of the EMNLP 2023*, pages 310–323. ACL.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 188–191. ACL.

Paul McNamee and James Mayfield. 2002. Entity extraction without language-specific resources. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Sudha Morwal, Nusrat Jahan, and Deepti Chopra. 2012. Named entity recognition using hidden markov model (hmm). *International Journal on Natural Language Computing (IJNLC) Vol*, 1.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Khaled Shaalan and Hafsa Raza. 2007. Person name entity recognition for arabic. In *Proceedings of the 2007 workshop on computational approaches to semitic languages: common issues and resources*, pages 17–24.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. Tener: adapting transformer encoder for named entity recognition. *arXiv preprint arXiv:1911.04474*.

Shaodian Zhang and Noémie Elhadad. 2013. Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. *Journal of biomedical informatics*, 46(6):1088–1098.