A NEW DATASET FOR NATURAL LANGUAGE UNDERSTANDING OF EXERCISE LOGS IN A FOOD AND FITNESS SPOKEN DIALOGUE SYSTEM

Maya Epps, Juan Uribe, Mandy Korpusik

Loyola Marymount University, Los Angeles, CA 90045, USA

ABSTRACT

Health and fitness are becoming increasingly important in the United States, as illustrated by the 70% of adults in the U.S. that are classified as overweight or obese, as well as globally, where obesity nearly tripled since 1975. Prior work used convolutional neural networks (CNNs) to understand a spoken sentence describing one's meal, in order to expedite the meal-logging process. However, the system lacked a complementary exercise-logging component. We have created a new dataset of 3,000 natural language exercise-logging sentences. Each token was tagged as an `Exercise`, `Feeling`, or `Other`, and mapped to the most relevant exercise, as well as a score of how they felt on a scale from 1 to 10. We demonstrate the following: for intent detection (i.e., logging a meal or exercise), logistic regression achieves over 99% accuracy on a held-out test set; for semantic tagging, contextual embedding models achieve 93% F1 score, outperforming conditional random field models (CRFs); and recurrent neural networks (RNNs) trained on a multiclass classification task successfully map tagged exercise and feeling segments to database matches. By connecting how the user felt while exercising to the food they ate, in the future we may provide personalized and dynamic diet recommendations.

***Index Terms*—** Semantic tagging, Bidirectional Encoder Representations from Transformers (BERT), RNN, CRF

1. INTRODUCTION

Obesity is a serious health concern in the United States and globally. In 2013, U.S. adults spent $80 billion in an attempt to lose weight.[1] The next year, the National Institute of Health reported that 175 million, or about 70%, of U.S. adults were "overweight or obese." [2] In 2018, 49% of U.S. adults were "trying to lose weight" [1]. "Worldwide obesity has nearly tripled since 1975," and there are more obese people than underweight [2]. "Globally, around 23% of adults" over 18 "were not active enough in 2010," and "insufficient physical activity is one of the leading" causes of death worldwide [3].

Healthy eating and physical exercise are important elements of weight control, but can be written off as simply too difficult or time-consuming. One contributing factor may be that current diet and exercise logging apps, such as MyFitnessPal, are tedious to use because the user has to manually enter and scroll through many food or exercise options to find the specific type of food or exercise the user wants to log.

Exercise	I	just	ran	three	miles	on	the	track
Tag:	O	O	BE	O	O	O	O	O

How they felt	I	'm	really	out	of	shape
Tag:	O	O	O	BF	IF	IF

Fig. 1. BIO tagging on a sample user utterance from the new corpus. The tags are `BE` (i.e., `Begin-Exercise`), `IE` (i.e., `Inside-Exercise`), `BF` (i.e., `Begin-Feeling`), `IF` (i.e., `Inside-Feeling`, and `O` (i.e., `Other`).

Prior work in the space of spoken diet tracking includes the spoken diet tracking system *Coco Nutritionist*, which makes food tracking much easier for adults trying to watch their diet [4, 5]. This system allows the user to speak out loud what they ate, and it parses the sentence in order to log the food that was mentioned. It does this by matching the words it recognizes as food to the U.S. Department of Agriculture (USDA) food database using deep neural networks [6, 7]. Specifically, the system uses a convolutional neural network (CNN) model to recognize these tags since this model is efficient, interpretable, and effective on this task.

The addition of an exercise logging component to *Coco Nutritionist* is needed because diet is only one aspect of a healthy lifestyle. Long-term benefits of exercise include weight control, a lower risk of stroke and heart disease, improved sleep quality, decreased risk of depression, and increased lifespan.[3] Creating an easy-to-use exercise logging component will encourage users to log their exercise more often. The system will log the exercise as it does food, by matching words the system recognizes to a database of exercises. Regular exercise logging may allow the user to create an exercise routine, improve upon previous routines in increments to build strength and endurance, and inspire

[1] https://money.usnews.com/money/personal-finance/articles/2013/01/02/the-heavy-price-of-losing-weight

[2] https://www.niddk.nih.gov/health-information/health-statistics/overweight-obesity

[3] https://www.hhs.gov/fitness/be-active/importance-of-physical-activity/index.html

the user to exercise more. With enough people logging their daily diet and exercise, we may even be able to find correlations, including what foods result in the best performance and energy levels for different demographics (e.g., youth, elderly, gluten-free diet, vegetarian diet). This would allow for providing personalized suggestions to the user based on their demographics and tracking history [8, 9].

This paper describes the key technical details of the novel exercise logging component in our food and fitness spoken dialogue system (see Figure 1), as well as a new, **publicly available dataset** of 3,000 natural language exercise logs with their semantic tag and database match annotations.[4] We show that statistical classifiers (specifically logistic regression and random forest) correctly predict the user intent with an accuracy over 99%. We then demonstrate that the recently released contextual embedding models such as Bidirectional Encoder Representations from Transformers (BERT) outperform prior state-of-the-art conditional random field (CRF) and long short-term memory (LSTM) on a semantic tagging task with 93% F1 score (i.e., harmonic mean of precision and recall) on a held-out test set. We demonstrate that recurrent networks map from exercise and feeling logs to the best database matches with top-1 recall scores of 77% and 80%, respectively (where we define top-1 recall as the number of test instances in which the correct exercise was ranked first).

2. RELATED WORK

Recently, neural networks such as bidirectional recurrent neural networks (RNNs) [10, 11], long short-term memory (LSTMs) [12], and convolutional neural networks (CNNs) [13], have been shown to outperform conditional random fields (CRFs) in spoken language understanding, which motivates the use of neural networks on our novel exercise understanding task. In addition, there has been work on jointly training RNNs for slot filling and intent and domain detection [14, 15, 16, 17, 18], as well as end-to-end neural networks for mapping directly from speech to semantic tags [19].

Within the past year, several papers have come out that learn *contextual* representations of sentences, where the entire sentence is used to generate embeddings. ELMo [20] uses a linear combination of vectors extracted from intermediate layer representations of a bidirectional LSTM trained on a large text corpus as a language model; in this feature-based approach, the ELMo vector of the full input sentence is concatenated with the standard context-independent token representations and passed through a task-dependent model for final prediction. This showed performance improvement over state-of-the-art on six NLP tasks, including question answering, textual entailment, and sentiment analysis. On the other hand, the OpenAI GPT [21] is a fine-tuning approach, where they first pre-train a multi-layer Transformer [22] as a language model on a large text corpus, and then conduct supervised fine-tuning on the specific task of interest, with a linear softmax layer on top of the pre-trained Transformer. Google's BERT [23] is a fine-tuning approach similar to GPT, but with the key difference that instead of combining separately trained forward and backward Transformers, they instead use a *masked* language model for pre-training, where they randomly masked out input tokens and predicted only those tokens. They demonstrated state-of-the-art performance on 11 NLP tasks, including the CoNLL 2003 named entity recognition task, which is similar to our semantic tagging task. Finally, many models have recently been developed that improve upon BERT, including RoBERTa (which improves BERT's pre-training by using bigger batches and more data) [24], XLNet (which uses Transformer-XL and avoids BERT's pretrain-finetune discrepancy through learning a truly bidirectional context via permutations over the factorization order) [25], and ALBERT (a lightweight BERT) [26].

3. DATA COLLECTION AND ANNOTATION

We collected three different types of data in one Amazon Mechanical Turk (AMT) task [27, 28]: exercise logs, tags, and values. The tasks were created and completed in Qualtrics due to the helpful tools it provides such as response-length checking and the ability to "pipe" previous responses into subsequent questions. First, the workers described a real or imaginary exercise they had performed, as well as how they felt during or after that exercise, in the same manner they would expect to describe them to a conversational agent (see Table 1 for a few examples). Next, the workers were asked to tag the specific words that described the exercise and how they felt. Finally, they were asked to assign a value to the identified words in the previous step. For the exercise identified, this value was an exercise from a predefined list of exercise words. The list was manually updated when it was missing exercises that the workers were logging. For the feeling logs, they were asked to assign their identified word or phrase a value on a scale from 1, meaning very bad and they would not want to feel this way during or after exercise again, to 10, meaning very good and they would want to feel that way during or after exercising again (see Figure 2). Due to data quality issues, we manually edited some of the responses that had more minor errors to ensure a high quality dataset.

During our first attempt at collecting data in this format on AMT, we realized that the quality of some of our responses was not as good as we had hoped. Some issues we identified included workers simply copying and pasting some of the examples instead of coming up with unique logs, incoherent logs that didn't make sense, misidentification of which words in the logs were the exercises or how they felt, and logs that were missing any description of an exercise or how they felt. To solve this problem, we ended up checking over most of the logs by hand, only accepting the logs that were com-

[4]Code and data at: https://github.com/mayaepps/exercise-logs

Exercise log	Exercise Value	Feeling Value
I **ran** a mile , I felt pretty **good** afterward . Much more **energetic** .	Running	6
I did **weight lifting** and felt much more **exhausted** but still **strong**	Dead Lifts	5
I performed 45 weighted **squats** , and I felt **pumped up** .	Squats	10
I **curled** 50lbs and my arms were **very sore** .	Bicep Curl	2
I **cycled** for one hour and my legs were in so much **pain** after !	Bicycling	2

Table 1. Five examples from the new exercise logging corpus, where the tagged exercise and feeling segments are in bold. The ground truth exercise and feeling labels are shown in the columns to the right.

pleted correctly, and correcting simple errors. For instance, in the example in Figure 1 above, the worker might have mislabeled the exercise as `Walking` or `Other` when their log specifically describes `Running`, or incorrectly identified the word or phrase that described the exercise as "miles" instead of "ran," or how they felt as "felt" instead of "terrible." In addition, to make this approval process much more efficient, we wrote our own checks that each part of the task had to pass in order to move on to the next part. For instance, some of these checks included making sure the word they identified as an exercise was also present verbatim and without mistakes in the exercise log, confirming the word or phrase they identified as how they felt was in their feeling log, and checking that the logs they entered were not in the examples that were given as a part of the instructions.

To measure the inter-annotator agreement, we computed Fleiss's Kappa score [29] as follows:

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \tag{1}$$

where $1 - \bar{P}_e$ is the possible agreement above chance, $\bar{P} - \bar{P}_e$ is the observed agreement above chance, and a kappa score of one indicates perfect agreement. $\bar{P}$ is the mean of each token's agreement, and $\bar{P}_e$ is the sum of squares for each category's proportion of words assigned to that category. We measured agreement for three types of per-token categories: semantic tags, exercise labels, and feeling labels (which we binarized to positive or negative sentiment). The kappa score is 0.53 for semantic tags, which indicates moderate agreement. Consistency among exercise and feeling labels is lower, with only fair agreement of 0.36 for exercise labels and 0.50 for binarized sentiment, indicating the task's difficulty.

In total, we collected 3,000 annotated exercise logs, each concatenated with their respective annotated feeling logs. 20% of that data (600 logs) constituted the test set, while the other 80% of the data (2,400 logs) made up the training set (Tables 2 and 3). This is a preliminary study on the new exercise logging corpus; in order to improve our database mapping NN models, which overfit to small datasets, we will perform data augmentation in future work.

Dataset	# Train Data	# Test Data	# Tags
Meal logs	35,130	3,412	5
Exercise logs	2,400	600	5

Table 2. The data statistics for each corpus.

BE	BI	BF	IF	O
3,037	1,040	3,830	1,184	32,302

Table 3. The frequency of each tag in the exercise corpus.

4. MODELS

Here we describe the full pipeline for our food and fitness spoken dialogue system, including the following three components for the new exercise logging component: user intent detection, semantic tagging of natural language descriptions, and database mapping to retrieve the results. The user's speech is first converted into text with the Google recognizer (if using the browser) or the Siri recognizer (if on iOS).

4.1. Intent Detection

In order to predict the user intent (i.e., whether they logged a meal or an exercise), we trained logistic regression and random forest classifiers on standard features: word counts; unigram, bigram, and trigram TFIDF scores; and character-level bigram and trigram TFIDF, where TFIDF represents the term frequency inverse document frequency.

4.2. Semantic Tagging

The second step in the spoken dialogue system for food and fitness is semantic tagging within the domain of interest. In this work, we focus on fitness. Thus, the tags are `B-Exercise`, `I-Exercise`, `B-Feeling`, `I-Feeling`, and `Other`, as shown in Figure 1. As in prior work [30], we compare previous state-of-the-art neural networks and conditional random field (CRF) models to newer contextual embedding methods such as BERT.

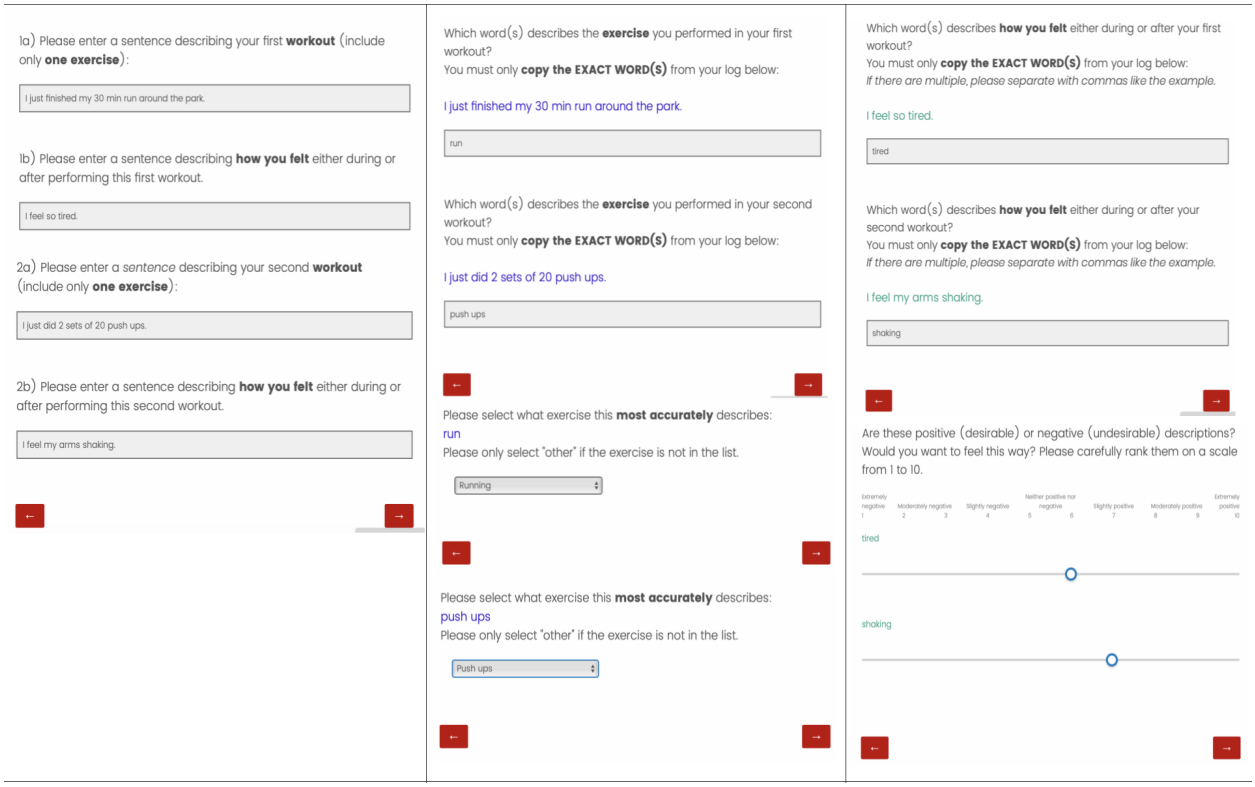

Fig. 2. The AMT task on Qualtrics for tagging and database mapping of exercise logs.

4.2.1. *Majority Baseline*

For the baseline, we predict the tag for each token that was assigned to it most often in the training data.

4.2.2. *Conditional Random Field*

The CRF features we use consist of the lowercase unigram; the suffix of the token; whether the token is all caps, a title, or a digit; whether the token is at the beginning or end of the sentence; the previous word; and the subsequent word. Although CRFs are a powerful discriminative classifier for sequential tagging problems, they require manual feature engineering, which is why we also investigated neural network models that do not require any manual feature engineering.

4.2.3. *Neural Network Models*

We implemented three baseline neural network models in the PyTorch deep learning toolkit [31]: a feed-forward (FF) network, a convolutional neural network (CNN) [30, 7], and a long short-term memory (LSTM) variant of RNNs. These models first feed the input exercise log through a learned embedding layer, followed by a single convolutional or recurrent layer, and a final linear layer with a softmax function in order to generate a probability distribution over all possible tags for each input token. In addition, we implemented a biLSTM-CRF, i.e., a CRF layer on top, where the emission features are taken from the hidden layer output of a bidirectional LSTM.

4.2.4. *Contextual Embedding Models*

Finally, we investigated four contextual embedding models that have been released over the past couple years, demonstrating success on many natural language processing tasks, including sequence labeling tasks such as ours. We used the base pre-trained BERT, XLNET, RoBERTa, and ALBERT models and tokenizers in PyTorch with a fine-tuned softmax token classification layer added on top specific to our domain.

4.3. Database Mapping

The last step is to rank the database matches, using as input either the entire exercise log or the tagged exercise or feeling segments, which we hypothesized would perform better.

4.3.1. *Exact String Matching Baseline*

First, we conducted a simple database lookup that found exact string matches, given the input sentence or segment. This approach clearly suffers when predicting how the user felt, since we map that to a number, but the user does not describe

in natural language their feelings with numbers. We thus need a semantic representation.

4.3.2. Embedding Similarity Baseline

The second baseline we implemented was predicting the best database match by ranking according to embedding cosine similarity scores. We summed the vectors for each lowercased token in either the segment or full description of the exercise and feeling log, using one of three pre-trained embeddings: word2vec [32], Glove [33], or FastText [34]. Since the feeling labels are numeric values rather than words, we used all the feeling segments in the training data as potential matches, and mapped to their assigned numeric value.

4.3.3. Logistic Regression Models

We formulate the problem as a multiclass classification problem, where the classifier directly predicts one of the database matches as the output. We used two types of input features: bag of words (BoW) and string similarity scores (i.e., Monge Elkan [35], Jaro Winkler [36], TFIDF, and soft TFIDF).

4.3.4. Neural Network Models

As in our prior work, we also trained neural network (NN) models on the multiclass task, which we found outperformed the binary verification approach used in prior work [7]. We fed as input to the network either the full user's exercise log, or only a tagged exercise (or feeling) segment. The architecture consisted of an input embedding layer (64-dim), a gated recurrent unit (GRU) with ReLU activation (128-dim), and a linear layer which output the logits of the final predictions.

5. EXPERIMENTS

5.1. Intent Detection

To balance the data, we used the full corpus of exercise logs and only a subset of 3,000 food logs. We then randomly shuffled and split the data into 90% training and 10% testing.

Model	Features	Accuracy
LR	Word counts	99.4
LR	Unigram TFIDF	**99.6**
LR	All n-gram + Char TFIDF + Counts	99.4
RF	Word counts	**99.5**
RF	Unigram TFIDF	99.4
RF	All n-gram + Char TFIDF + Counts	99.2

Table 4. Intent detection accuracy with logistic regression (LR) and random forest (RF) classifiers using various feature sets, averaged over three runs on the held-out test set.

Interestingly, we see in Table 4 that simpler feature sets (i.e., word counts or unigram TFIDF scores) seem to work best for intent detection, rather than adding n-grams or character n-grams. This may be due to the data's simplicity, since it is possible for the model to distinguish between the two intents using one word only (e.g., a food name or an exercise).

5.2. Semantic Tagging

We split the training data into training and validation for fine-tuning the neural network hyperparameters. We experimented with both pre-trained embeddings (i.e., Glove [33], word2vec [32], and fastText [34]) and learning embeddings from scratch, adding multiple layers, lowercasing, and sweeping the hidden dimension size. We found that lowercasing helped, and that using pre-trained embeddings helped the LSTM, but not the CNN or FF network. Thus, for the experiments shown in Table 5, we used pre-trained word2vec embeddings for the LSTM, and learned embeddings from scratch for the CNN and FF network. We used one hidden layer per model. The hidden dimension was 64 for FF and 256 for LSTM. The embedding dimension was 50 when trained from scratch and 300 for word2vec. For the CNN, we used 64 filters of width 1, 2, 3, and 5. We used the SGD optimizer, negative log-likelihood loss, and 0.1 learning rate.

For the contextual embeddings, we used the base cased models (they outperformed uncased) and default hyperparameters of a batch size of 32 and fine-tuning for 3 epochs [37].

Model	BE	BF	IE	IF	O	Avg
Majority	0.75	0.78	0.56	0.23	0.94	0.88
CRF	0.85	0.84	0.66	0.46	0.96	0.91
BiLSTM-CRF	0.81	0.79	0.60	0.44	0.95	0.90
FF	0.80	0.77	0.55	0.31	0.95	0.89
CNN	0.80	0.75	0.56	0.23	0.95	0.89
LSTM	0.85	0.82	0.61	0.56	0.96	0.91
BERT	0.90	0.86	0.74	0.52	**0.97**	**0.93**
RoBERTa	**0.91**	0.86	0.74	**0.58**	**0.97**	**0.93**
XLNet	0.91	**0.87**	0.73	0.51	**0.97**	**0.93**
ALBERT v2	0.90	0.86	**0.77**	0.55	**0.97**	**0.93**

Table 5. Semantic tagging F1 scores for several neural models, a CRF, and a majority baseline. The tags are `BE (Begin-Exercise)`, `BF (Begin-Feeling)`, `IE (Inside-Exercise)`, `IF (Inside-Feeling)`, `O (Other)`, and the weighted average of all tags.

As shown in Table 5, the majority baseline performs the worst. It often mislabels common words such as "of," and "to" as `Other` (O) since that is usually the correct tag. However, these should sometimes be part of the feeling description (IF), as shown in Figure 3 below. For completeness, we also report entity-level scores in Table 6, as is standard in

the CoNLL (Conference on Computational Natural Language Learning) tasks for named entity recognition [38].

Model	Exercise F1	Feeling F1	Avg F1
Majority	64.1	67.2	65.8
CRF	78.1	80.3	79.3
BiLSTM-CRF	73.5	76.2	74.9
FF	69.5	68.3	68.9
CNN	68.5	68.1	68.3
LSTM	73.7	75.1	74.5
BERT	79.0	**82.3**	80.8
RoBERTa	79.4	82.0	80.9
XLNet	**80.7**	82.0	**81.4**
ALBERT v2	79.0	**82.3**	80.8

Table 6. F1 scores per exercise and feeling *entity*. With CoNLL-style evaluation, performance differences are clearer.

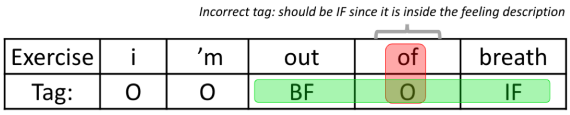

Fig. 3. An example error by the majority baseline.

All the models are best at predicting the `Other` tag (O), since this tag occurs most often. The second easiest tags are `Begin-Exercise` (BE) and `Begin-Feeling` (BF), whereas `Inside-Exercise` (IE) and `Inside-Feeling` (IF) are most difficult due to appearing the least often. The CRF and LSTM outperform the FF and CNN, but all the contextual embeddings perform by far the best, with a weighted F1 score of 93% (averaged over all semantic tags):

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (2)$$

In general, performance on tagging exercise segments seems to be higher than that on feeling segments.

5.3. Database Mapping

We evaluated the database mapping with top-1 recall scores (i.e., the number of test instances in which the correct exercise was ranked first), for random and string matching baselines, logistic regression, and neural networks, as shown in Table 7 for exercise, and in Table 8 for feelings. We used an SGD optimizer with a learning rate of 0.01 and momentum of 0.9.

We found that predicting how the user felt is harder than for exercise, since we are mapping from natural language to a number (even when we reduce the 10 classes to two, i.e., binary). The GRU outperformed an LSTM, likely due to our small dataset, for which an LSTM may be too powerful. For exercise, we confirmed our hypothesis that mapping from the tagged segment performs better than using the whole sentence as input. These are preliminary results—we will collect more data so we can properly train an LSTM without overfitting.

Model	Input Sentence	Exercise Segment
Random	0.01	0.01
Str matching	0.41	0.66
Glove emb sim	0.10	0.58
w2v emb sim	0.05	0.42
FastText emb sim	0.07	0.47
LR BoW	**0.69**	0.74
GRU	0.60	**0.77**

Table 7. Top-1 recall scores for *exercise* mapping, with either the whole sentence as input, or only the exercise segment.

Model	Input Sentence	Feeling Segment
Random	0.10	0.10
Str matching	0.07	0.05
LR BoW	0.28	0.28
GRU	0.28	0.28
Random (binary)	0.50	0.50
Glove (binary)	0.65	0.70
w2v (binary)	0.69	0.73
FastText (binary)	0.67	0.72
GRU (binary)	**0.80**	**0.80**

Table 8. Top-1 recall scores for *sentiment* mapping, with either the whole sentence as input, or only the feeling segment.

6. CONCLUSION

In this paper, we have illustrated the success of deep neural networks for incorporating a novel exercise logging component into an existing food and fitness spoken dialogue system. We have shown that, in particular, contextual embeddings such as XLNet outperform prior state-of-the art CRFs and LSTMs on semantic tagging, without requiring any manual feature engineering or hyperparameter fine-tuning. The ease with which such models can be ported to a new domain, and their superior performance on a wide array of natural language processing tasks, demonstrates the importance of pretraining neural network models on large datasets.

In future work, we plan to collect more data, especially spoken exercise logging data in the wild on the iOS platform *Coco Nutritionist*, to further refine our models. With sufficient users and data, we may learn correlations between diet and how users felt while exercising, enabling us to provide personalized recommendations. Finally, we plan to incorporate data augmentation techniques for training neural models for the database mapping component, specifically Noisy Student [39] and Psuedo Meta Labels [40].

7. REFERENCES

[1] Jamie Ducharme, "About half of Americans say they're trying to lose weight," https://time.com/5334532/weight-loss-americans, 2018.

[2] World Health Organization, "Obesity and overweight," https://www.who.int/news-room/fact-sheets/detail/obesity-and-overweight, 2020.

[3] World Health Organization, "Physical activity," https://www.who.int/news-room/fact-sheets/detail/physical-activity, 2020.

[4] M. Korpusik, N. Schmidt, J. Drexler, S. Cyphers, and J. Glass, "Data collection and language understanding of food descriptions," *Proceedings of 2014 IEEE Spoken Language Technology Workshop (SLT)*, pp. 560–565, 2014.

[5] M. Korpusik and J. Glass, "Spoken language understanding for a nutrition dialogue system," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 1450–1461, 2017.

[6] M. Korpusik, Z. Collins, and J. Glass, "Semantic mapping of natural language input to database entries via convolutional neural networks," *Proceedings of IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5685–5689, 2017.

[7] Mandy Korpusik and Jim Glass, "Deep learning for database mapping and asking clarification questions in dialogue systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019.

[8] M. Korpusik and J. Glass, "Convolutional neural networks and multitask strategies for semantic mapping of natural language input to a structured database," in *Proceedings of IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6174–6178.

[9] Mandy B Korpusik, *Deep learning for spoken dialogue systems: application to nutrition*, Ph.D. thesis, Massachusetts Institute of Technology, 2019.

[10] G. Mesnil, X. He, L. Deng, and Y. Bengio, "Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding.," in *Proceedings of the Fourteenth International Conference on Spoken Language Processing (Interspeech)*, 2013, pp. 3771–3775.

[11] G. Mesnil, Y. Dauphin, K. Yao, Y. Bengio, L. Deng, D. Hakkani-Tur, X. He, L. Heck, G. Tur, D. Yu, et al., "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 530–539, 2015.

[12] K. Yao, B. Peng, Y. Zhang, D. Yu, G. Zweig, and Y. Shi, "Spoken language understanding using long short-term memory neural networks," in *Proceedings of the 2014 IEEE Workshop on Spoken Language Technology (SLT)*. IEEE, 2014, pp. 189–194.

[13] P. Xu and R. Sarikaya, "Convolutional neural network based triangular CRF for joint intent detection and slot filling," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2013, pp. 78–83.

[14] D. Hakkani-Tür, G. Tür, A. Celikyilmaz, Y. Chen, J. Gao, L. Deng, and Y. Wang, "Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM," in *Interspeech*, 2016, pp. 715–719.

[15] J. Lee, D. Kim, R. Sarikaya, and Y. Kim, "Coupled representation learning for domains, intents and slots in spoken language understanding," *arXiv preprint arXiv:1812.06083*, 2018.

[16] B. Liu and I. Lane, "Attention-based recurrent neural network models for joint intent detection and slot filling," *arXiv preprint arXiv:1609.01454*, 2016.

[17] M. Ma, K. Zhao, L. Huang, B. Xiang, and B. Zhou, "Jointly trained sequential labeling and classification by sparse attention neural networks," *arXiv preprint arXiv:1709.10191*, 2017.

[18] C. Goo, G. Gao, Y. Hsu, C. Huo, T. Chen, K. Hsu, and Y. Chen, "Slot-gated modeling for joint slot filling and intent prediction," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 2 (Short Papers)*, 2018, vol. 2, pp. 753–757.

[19] P. Haghani, A. Narayanan, M. Bacchiani, G. Chuang, N. Gaur, P. Moreno, R. Prabhavalkar, Z. Qu, and A. Waters, "From audio to semantics: Approaches to end-to-end spoken language understanding," *arXiv preprint arXiv:1809.09190*, 2018.

[20] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[21] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding

by generative pre-training," *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*, 2018.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 5998–6008.

[23] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[24] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[25] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in neural information processing systems*, 2019, pp. 5754–5764.

[26] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[27] M. Korpusik, C. Huang, M. Price, and J. Glass, "Distributional semantics for understanding spoken meal descriptions," *Proceedings of 2016 IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6070–6074, 2016.

[28] M. Korpusik and J. Glass, "Dialogue state tracking with convolutional semantic taggers," in *Proceedings of IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.

[29] A. Viera, J. Garrett, et al., "Understanding interobserver agreement: The kappa statistic," *Family Medicine*, vol. 37, no. 5, pp. 360–363, 2005.

[30] Mandy Korpusik, Zoe Liu, and James Glass, "A comparison of deep learning methods for language understanding," *Proc. Interspeech 2019*, pp. 849–853, 2019.

[31] N. Ketkar, "Introduction to pytorch," in *Deep learning with python*, pp. 195–208. Springer, 2017.

[32] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, L Sutskever, and G Zweig, "word2vec," *URL https://code. google. com/p/word2vec*, 2013.

[33] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," *Proceedings of 2014 Conference on Empirical Methods on Natural Language (EMNLP)*, vol. 12, 2014.

[34] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[35] Alvaro Monge and Charles Elkan, "An efficient domain-independent algorithm for detecting approximately duplicate database records," 1997.

[36] Matthew A Jaro, "Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida," *Journal of the American Statistical Association*, vol. 84, no. 406, pp. 414–420, 1989.

[37] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew, "Huggingface's transformers: State-of-the-art natural language processing," *ArXiv*, vol. abs/1910.03771, 2019.

[38] Erik F Sang and Sabine Buchholz, "Introduction to the CoNLL-2000 shared task: Chunking," *arXiv preprint cs/0009008*, 2000.

[39] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le, "Self-training with noisy student improves ImageNet classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10687–10698.

[40] Hieu Pham, Qizhe Xie, Zihang Dai, and Quoc V Le, "Meta pseudo labels," *arXiv preprint arXiv:2003.10580*, 2020.