INFORMS Journal on Data Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

HeBERT and HebEMO: A Hebrew BERT Model and a Tool for Polarity Analysis and Emotion Recognition

Avihay Chriqui, Inbal Yahav

To cite this article:
Avihay Chriqui, Inbal Yahav (2022) HeBERT and HebEMO: A Hebrew BERT Model and a Tool for Polarity Analysis and Emotion Recognition. INFORMS Journal on Data Science 1(1):81-95. https://doi.org/10.1287/ijds.2022.0016

http://pubsonline.informs.org/journal/ijds

INFORMS JOURNAL ON DATA SCIENCE
Vol. 1, No. 1, April–June 2022, pp. 81–95
ISSN 2694-4022 (print), ISSN 2694-4030 (online)

HeBERT and HebEMO: A Hebrew BERT Model and a Tool for Polarity Analysis and Emotion Recognition

Avihay Chriqui,[a,*] **Inbal Yahav**[a]

[a] Coller School of Management, Tel Aviv University, Tel Aviv 6997801, Israel
*Corresponding author
Contact: avichayc@mail.tau.ac.il, https://orcid.org/0000-0003-2498-377X (AC); inbalyahav@tauex.tau.ac.il, https://orcid.org/0000-0002-1513-017X (IY)

Received: March 25, 2021
Revised: February 13, 2022
Accepted: April 1, 2022
Published Online in Articles in Advance: June 7, 2022

https://doi.org/10.1287/ijds.2022.0016

Abstract. Sentiment analysis of user-generated content (UGC) can provide valuable information across numerous domains, including marketing, psychology, and public health. Currently, there are very few Hebrew models for natural language processing in general, and for sentiment analysis in particular; indeed, it is not straightforward to develop such models because Hebrew is a morphologically rich language (MRL) with challenging characteristics. Moreover, the only available Hebrew sentiment analysis model, based on a recurrent neural network, was developed for polarity analysis (classifying text as positive, negative, or neutral) and was not used for detection of finer-grained emotions (e.g., anger, fear, or joy). To address these gaps, this paper introduces HeBERT and HebEMO. HeBERT is a transformer-based model for modern Hebrew text, which relies on a BERT (bidirectional encoder representations from transformers) architecture. BERT has been shown to outperform alternative architectures in sentiment analysis and is suggested to be particularly appropriate for MRLs. Analyzing multiple BERT specifications, we find that whereas model complexity correlates with high performance on language tasks that aim to understand terms in a sentence, a more parsimonious model better captures the sentiment of an entire sentence. Notably, regardless of the complexity of the BERT specification, our BERT-based language model outperforms all existing Hebrew alternatives on all language tasks examined. HebEMO is a tool that uses HeBERT to detect polarity and extract emotions from Hebrew UGC. HebEMO is trained on a unique COVID-19-related UGC data set that we collected and annotated for this study. Data collection and annotation followed an active learning procedure that aimed to maximize predictability. We show that HebEMO yields a better performance accuracy for polarity classification. Emotion detection reaches high performance for various target emotions, with the exception of surprise, which the model failed to capture. These results are better than the best reported performance, even among English-language models of emotion detection.

Funding: Financial support from the Jeremy Coller Foundation, Tel Aviv University [Grant 08120001000], and the Henry Crown Center for Business Research is gratefully acknowledged.
History: Shawndra Hill served as the senior editor for this article.
Data Ethics & Reproducibility Note: The data for this paper were collected according to accepted ethical standards. The code capsule is available on Code Ocean at https://doi.org/10.24433/CO.3045134.v1 and in the e-Companion to this article (available at https://doi.org/10.1287/ijds.2022.0016).

Keywords: HeBERT • Hebrew NLP • sentiment analysis • polarity analysis • emotion recognition

1. Introduction

Sentiment analysis, also referred to as opinion mining or subjectivity analysis (Liu and Zhang 2012), is probably one of the most common tasks in natural language processing (NLP) (Liu 2012, Zhang et al. 2018). The goal of sentiment analysis is to systematically extract from written text what people think or feel toward entities such as products, services, individuals, events, news articles, and topics.

Sentiment analysis includes multiple types of tasks, one of the most common being *polarity* classification, the binning of overall sentiment into the three categories of positive, neutral, or negative. Another prominent sentiment analysis task is *emotion detection*, a process for extracting finer-grained emotions such as happiness, anger, and fear from human language. These emotions, in turn, can shed light on individuals' beliefs, behaviors, or mental states.

Both polarity classification and emotion detection have proven to yield valuable information in diverse applications. Research in marketing, for example, has shown that emotions that users express in online product reviews affect products' virality and profitability (Chitturi et al. 2007, Ullah et al. 2016, Adamopoulos et al. 2018). In finance, Bellstam et al. (2020) extracted sentiments from financial analysts' textual descriptions of firm activities and used those sentiments to measure corporate innovation. In psychology, sentiment

analysis has been used to detect distress in psychotherapy patients (Shapira et al. 2020) and to identify specific emotions that might be indicative of suicidal intentions (Desmet and Hoste 2013). Notably, recent studies suggest that the capacity to identify certain emotions (e.g., fear or distress) can contribute to the understanding of individuals' behaviors and mental health in the COVID-19 pandemic (Ahorsu et al. 2020, Pfefferbaum and North 2020).

The literature offers a considerable number of methods and models for sentiment analysis, with a strong bias toward polarity detection. Models for emotion detection, though less common, are also accessible to the research community in multiple languages. As yet, however, emotion detection models do not support the Hebrew language. In fact, to our knowledge, only one study thus far has developed a Hebrew-language model for sentiment analysis of any kind (specifically, polarity classification; Amram et al. 2018). Notably, existing sentiment analysis methods developed for other languages are not easily adjustable to Hebrew because of the unique linguistic and cultural features of this language.

A key challenge in the development of Hebrew-language sentiment analysis tools relates to the fact that Hebrew is a morphologically rich language (MRL), defined as a language "in which significant information concerning syntactic units and relations is expressed at word-level" (Tsarfaty et al. 2010, pp. 1–12). In Hebrew, as in other MRLs (e.g., Arabic), grammatical relations between words are expressed via the addition of affixes (suffixes, prefixes), instead of the addition of particles. Moreover, the word order in Hebrew sentences is rather flexible. Many words have multiple meanings, which change depending on context. Furthermore, written Hebrew contains vocalization diacritics, known as *Niqqud* ("dots"), which are missing in nonformal scripts. Other Hebrew characters represent some, but not all, of the vowels. Thus, it is common for words that are pronounced differently to be written in the same way. These unique characteristics of Hebrew pose a challenge in developing appropriate Hebrew NLP models. Architectural choices should be made with care, to ensure that the features of the language are well represented. The current best practice for Hebrew NLP is the use of the multilingual BERT model (mBERT, based on the BERT (bidirectional encoder representations from transformers) architecture, discussed further below; Devlin et al. 2018), which was trained on a small size Hebrew dictionary. When tested on Arabic (the closest language to Hebrew), mBERT was shown to have significantly lower performance than a language-specific BERT model on multiple language tasks (Antoun et al. 2020).

This paper achieves two main goals related to the development of Hebrew-language sentiment analysis capabilities. First, we pretrain a language model for modern Hebrew, called HeBERT, which can be implemented in diverse NLP tasks, and is expected to be particularly appropriate for sentiment analysis (as compared with alternative model architectures). HeBERT is based on the well-established BERT architecture (Devlin et al. 2018). The latter was originally trained for the unsupervised fill-in-the-blank task (known as masked language modeling; Fedus et al. 2018). We train HeBERT on two large-scale Hebrew corpuses: Hebrew Wikipedia and OSCAR (Open Superlarge Crawled ALMAnaCH corpus, a huge multilingual corpus based on open web crawl data; Ortiz Suárez et al. 2020). We then evaluate HeBERT's performance on five key NLP tasks, namely, fill in the blank, out-of-vocabulary (OOV), named-entity recognition (NER), part of speech (POS), and sentiment (polarity) analysis. We examine several architectural choices for our model and put forward and test hypotheses regarding the relative performance of each alternative, ultimately selecting the best-performing option. Specifically, we show that whereas model complexity correlates with high performance on language tasks that aim to understand terms in a sentence, a more parsimonious model better captures the sentiment of an entire sentence.

Second, we develop a tool to detect sentiments—specifically, polarity and emotions—from user-generated content (UGC). Our sentiment detector, called HebEMO, is based on HeBERT and operates on a document level. We apply HebEMO to user-generated comments, from three major news sites in Israel, that were posted in response to COVID-19-related articles during 2020.[1] We chose this data set on the basis of findings that the COVID-19 pandemic intensified emotions in multiple communities (Pedrosa et al. 2020), suggesting that online discourse regarding the pandemic is likely to be highly emotional. Comments were selected for annotation following an innovative semisupervised active learning approach that aimed to maximize predictability.

We show that HebEMO achieves high performance on polarity classification, with a weighted average F_1 score of 0.96. Emotion detection reaches F_1 scores of 0.80–0.96 for the various target emotions, with the exception of surprise, which the model failed to capture (F_1 = 0.62). These results are better than the best reported performance, even when compared with English-language models for emotion detection (Mohammad et al. 2018, Ghanbari-Adivi and Mosleh 2019). We have made our models available online for PyTorch and Tensorflow implementations.[2] Examples of how to use our models can be found in Online Appendix A.

The remainder of this paper is organized as follows. In the next section, we provide a brief overview of the state of the art (SOTA) in sentiment analysis in general and emotion recognition in particular, and we also briefly discuss considerations that must be taken into

account when developing pretrained language models for sentiment analysis. Next, we present HeBERT, our language model, elaborating on how we address some of the unique challenges associated with the Hebrew language. We subsequently describe HebEMO and evaluate its performance on our UGC data.

2. Background

2.1. Language Specificity in Emotional Expression

Psychologists and psychoanalysts have long known that, despite the importance of nonverbal behavior, words are the most natural way to externally express an inner emotional world (Ortony et al. 1987). In line with this premise, theories of emotions stress that emotional experience and its intensity can be inferred from spoken or written language (Argaman 2010). Yet, emotions vary across cultures (Rosaldo et al. 1984), and, consequently, languages differ in the degree of emotionality they convey and in the ways in which emotions are expressed in words (Wierzbicka 1994, Kövecses 2003). In particular, as noted by Kövecses (2003), the verbalization of emotions commonly relies on the use of metaphorical and metonymic expressions, which may differ across languages. Religion is another source of variation in emotional experience and its associated expression (Kim-Prieto and Diener 2009). One study showed how the moral system of a culture—and, specifically, a Middle Eastern culture—can be linked to certain types of emotions, and suggested that differences in culturally dominant emotions can play a decisive role in cultural clashes (Fattah and Fierke 2009).

The above discussion implies that emotion detection tools that are implemented in one language might not be easily transferable to other languages, particularly languages that are culturally distant. Accordingly, sentiment analysis tools must be tailored to specific language models in order to provide informative results. The current paper proposes two such tools for the Hebrew language—tools that take into account specific linguistic challenges associated with Hebrew, elaborated in subsequent sections.

2.2. Overview of Sentiment Analysis Approaches

Many studies offer comprehensive overviews of common sentiment analysis methods (e.g., Hemmatian and Sohrabi 2019, Liu et al. 2019a, Yue et al. 2019, Yadav and Vishwakarma 2020). We present here the main points, with an emphasis on models that form the basis of this study. Most of the models described below were developed primarily for polarity analysis; however, as noted in the following subsection, the architectures are applicable to other sentiment analysis tasks such as emotion detection.

Current reviews on sentiment analysis tend to categorize the various approaches according to the granularity level of text that they accommodate (Liu et al. 2019a): *document level*, that is, evaluating whether an entire document expresses a particular type of sentiment (e.g., positive or negative); *sentence level*, that is, assigning a sentiment to each sentence in the document separately; and *aspect level*, that is, assigning sentiment to each "aspect" discussed in the text. The latter requires a preprocessing step to extract aspects from a written text. In this paper, we follow a document-level approach, elaborated further below.

Sentiment classification approaches can further be categorized into four main groups, according to their underlying methodologies. The first methodology is the lexicon-based approach. Based on the theory of emotions, this approach uses sentiment terms to score emotions in an input text. Linguistic Inquiry and Word Count, for example, is a popular software program that was developed to assess (among other features) emotions in text, using a psychometrically validated internal dictionary (Pennebaker et al. 2001). The main advantage of the lexicon-based approach is that it is *unsupervised*, meaning that it can be applied without any training or labeled data (Yue et al. 2019). The main limitation of this approach is that is does not account for the context of terms in the lexicon, and thus overlooks complex linguistic features such as sarcasm, ambiguity, and idioms (Liu 2012). Accordingly, its accuracy is fairly low compared with the alternative approaches.

The second sentiment analysis approach is supervised machine learning (ML)-based text classification. ML-based sentiment analysis commonly involves four main steps: (1) text preprocessing via standard procedures (e.g., converting to lowercase, removing stop words); (2) feature extraction, using techniques such as bag of words (e.g., Liu 2012, Mudinas et al. 2012), term frequency–inverse document frequency (TF-IDF)–adjusted bag of words (e.g., El-Din 2016, Luo et al. 2016), *n*-grams (e.g., Tripathy et al. 2016, Mughaz et al. 2018), and word embedding techniques such as the well-known word2vec model by Mikolov et al. (2013) (e.g., Tang et al. 2014); (3) feature engineering, including the addition of POS tags (e.g., Abdul-Mageed et al. 2011, Liu 2012, Wang et al. 2015, Mughaz et al. 2018), emoticons and emotion word extraction (e.g., Hogenboom et al. 2013, Yamamoto et al. 2014, Khan et al. 2016), and other domain-specific features (e.g., Abdul-Mageed et al. 2011); and (4) fitting a classification model to a set of labeled training data. The most common classification models used for ML-based sentiment analysis are support vector machines and naive base classification, because of their ability to handle high-dimensional space (Liu et al. 2013, Luo et al. 2016, Ren et al. 2016). Although the performance of these

methods is better than that of lexicon-based approaches, they are considered domain specific, and as such have limited generalizability (Yue et al. 2019).

The third sentiment classification approach is deep learning (DL) based. DL approaches are supervised methods that are based on multiple-layer neural networks. DL-based sentiment classification models differ by their network architecture. Common architectures include the following: (1) *convolutional neural networks* (*CNNs*), which transform a structured input layer (e.g., sentences or documents represented as bag-of-words or word-embedding vectors), via convolutional layers, into a sentiment class (Kim 2014); (2) *recursive or recurrent neural networks* (*RNNs*), which handle unstructured sequential data, such as textual sentences, and learn the relations between the sequential elements (Dong et al. 2014); and (3) *long short-term memory* (*LSTM*), a popular variant of RNN, which can catch long-term dependencies between data segments, in one direction (e.g., left to right) or in both (referred to as bidirectional LSTM, or BiLSTM; Hochreiter and Schmidhuber 1997).

In a recent paper, Amram et al. (2018) raised the question of the relationship between the characteristics of a language and the DL architectural choices of a sentiment classifier. They analyzed this question for the morphologically rich Hebrew language. Specifically, they compared the performance of CNN and BiLSTM architectures on a polarity classification task. They assumed that the latter method would implicitly capture main morphological signatures, and thus outperform the former. Interestingly, and in contrast to findings in English sentiment analysis (Yin et al. 2017, Acheampong et al. 2020), they found that CNN yielded overall better performance (accuracy of 0.89) than BiLSTM, even when the latter was trained on morphologically segmented inputs. As far as we know, this is the only paper that developed and evaluated a sentiment analysis model for the Hebrew language.

The last sentiment classification method, which we adopt in this paper, is the transfer learning-based approach. Transfer learning is the act of carrying knowledge gained from one problem and applying it to another, similar problem (Pan and Yang 2009). In NLP, transfer learning is implemented via *transformers* (Tay et al. 2020). Similarly to RNN, transformers use a DL approach to process sequential data. The primary advantage of the transformer is its unique attention mechanism, which eliminates the need to process data in order, and allows for parallelization (Vaswani et al. 2017). With transformers, a target language is first algorithmically learned, irrespective of the target language task (e.g., sentiment analysis task). To this end, a language model is trained on a preselected unsupervised NLP task (see Section 3 for details). Then the language model is transferred to the target task. This process is called *fine-tuning*.

Various pretrained language models have been used in transfer learning for NLP; these include fastText (Joulin et al. 2016), ELMo (Embeddings from Language Models, based on forward and backward LSTMs; Peters et al. 2018), GPT (Generative Pretrained Transformer) (Radford et al. 2018), and BERT (Devlin et al. 2018). Of these, BERT is one of the most common transformer models for NLP. For sentiment analysis tasks, BERT models—and transformer models in general—are widely used and produce the best results compared with alternatives (Zampieri et al. 2019, Patwa et al. 2020). For the Hebrew language, the only BERT model available is the multilingual BERT, mBERT (Devlin et al. 2018), which was trained on a small-sized Hebrew dictionary (about 2,000 tokens). Notably, for the Arabic language, which is the closest MRL to Hebrew, Antoun et al. (2020) showed that a pretrained Arabic BERT model achieved better performance on polarity analysis than did any other architecture (an improvement of 1%–6% in accuracy). The Arabic-specific model also achieved better performance compared with mBERT.

2.3. Emotion Recognition

Emotion recognition is a subtask in sentiment analysis that offers a finer granularity sentiment level compared with polarity analysis. Two definitions of human emotions dominate the NLP literature, with no clear preference between them (Kratzwald et al. 2018). The first definition, based on a theory developed by Ekman (1999), considers emotions as distinct categories, meaning that each emotion differs from the others in important ways rather than simply their intensity. Ekman (1999) identified six basic emotions, consistent across cultures, that fit facial expressions: anger, disgust, fear, happiness, sadness, and surprise. The second definition is based on a theory by Plutchik (1980), who stressed that emotions can be treated as dimensional constructs, and that there are relations between occurrences and intensities of basic emotions. In particular, Plutchik (1980) defined a "wheel" comprising four polar pairs of basic emotions: joy–sadness, anger–fear, trust–disgust, and surprise–anticipation. Combinations of dyads or triads of emotions define another set of 56 emotions. For example, envy is a combination of sadness and anger. This wheel serves as the theoretical basis of common automated emotion detection algorithms (Medhat et al. 2014). Notably, for the purpose of emotion detection, the two conceptualizations of emotion are generally compatible with each other, as they agree on the set of emotions defined as "basic" emotions.

Though common, emotion recognition is not as widespread as polarity analysis, and it is considered more challenging (Acheampong et al. 2020). A key challenge is that, whereas any text can be classified according to its polarity, not all texts contain emotions, and thus it is harder to infer emotions via a lexicon-based approach. This challenge is further compounded by the fact that labeled data are commonly not available. Furthermore, existing data sets are rather imbalanced. Naturally, the lack of data availability is more severe in non-English languages (Ahmad et al. 2020).

In general, the emotion detection task is treated as a multilabel classification task, and models for emotion recognition are similar in architecture to polarity detection models. Recent research has shown that in emotion detection tasks, pretrained BiLSTM architectures provide advantages over CNN and unidirectional RNN models (Acheampong et al. 2020), and that transformers are preferable to other DL approaches (Chatterjee et al. 2019, Zhong et al. 2019). For example, in a recent SemEval competition (Chatterjee et al. 2019) that included an emotion detection task for three emotions (angry, happy, and sad), transformer-based models were shown to give the best performance (performance ranges: F_1 = 0.75–0.8; precision, 0.78–0.85; recall, 0.78–0.85).

2.4. Training Language Models for Transfer Learning

As noted above, transfer learning for polarity analysis and/or emotion recognition requires a pretrained language model. To develop and train a language model, one needs to make the following three basic decisions:

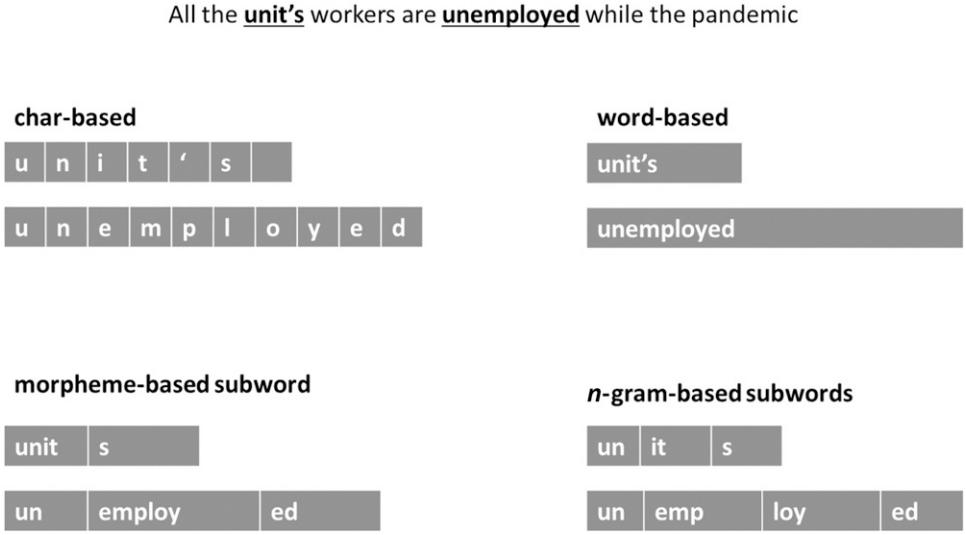
1. *Input representation (tokenization)*: What is the granularity of the tokens that are fed to the model? Common granularity levels include characters, *n*-gram-based subwords (derived using the WordPiece algorithm; Schuster and Nakajima 2012), morpheme-based subwords, and full words (see Figure 1 for the differences between the approaches).

2. *Architectural choices*: What is the exact architecture and specification of the neural network?

3. *Output*: What is the (unsupervised) task that the model is trained on?

Regarding the input representation, the choice of representation affects the features that the language model is able to capture and the training complexity. Character-based representation is better for learning word morphology, especially for low-frequency words and MRLs (Belinkov et al. 2017, Vania et al. 2018), but it comes with longer training times and a deeper architecture compared with other representations (Bojanowski et al. 2015). Word-based representation, in turn, treats each word as a separate token, and thus is considered better for understanding semantics (Pota et al. 2019). With this representation, however, words that differ by prefix or suffix are considered different, necessitating storage of a very large vocabulary. Moreover, OOV tokens are not represented. The intermediate option is to use a subword representation, which provides some balance between the character- and word-based representations; moreover, it overcomes the OOV problem associated with the word-based representation, and its vocabulary requirements are more manageable (Wu et al. 2016). With subwords, words can be broken either into *n*-gram characters or according to morphemes that have lingual meaning (but also higher computational costs). Previous literature has produced mixed results regarding to the extent to which using a morpheme-based approach can improve upon the

Figure 1. Input Representation Alternatives



n-gram-based approach (Bareket and Tsarfaty 2021). Recently, Klein and Tsarfaty (2020) showed that subword splitting in mBERT is suboptimal for capturing morphological information.

For the question of *architecture selection*, Devlin et al. (2018) and Radford et al. (2019) showed that for *similar model size*, BERT outperforms other architectures such as GPT and ELMo on sentiment tasks.

With respect to the model output, there are several commonly used tasks on which a model can be trained. The first is *predict the future*, meaning that the model is trained to predict the last token of a sentence. This task accounts for unidirectional contexts only. The second is the fill-in-the-blank task, where the model is trained to fill in a missing token within a sentence. This task takes into account the full (bidirectional) sentence context and is able to better capture the meanings of tokens, both syntactically and semantically (Devlin et al. 2018). Recently, Levine et al. (2020) offered a method to optimize these tasks, called *pointwise-mutual-information* (*PMI*) *masking*. The authors suggested that instead of filling in a single random token, the model should be trained to fill in a set of tokens that carry mutual information. Last, *next-sentence prediction* (*NSP*) has also been proposed as a training task (Devlin et al. 2018). Here, the model receives a pair of sentences taken from a larger text and is trained to predict whether one sentence follows the other in that text. However, recent empirical studies have raised questions regarding the effectiveness of this task for transfer learning. Specifically, Liu et al. (2019b) showed that models that were trained on blocks of text from a single document, rather than on pairs of single sentences (as required in NSP), achieved better performance on downstream tasks.

3. HeBERT: Language Model

In this section, we develop an unsupervised Hebrew BERT model, which we will later fine-tune for the tasks of polarity analysis and emotion recognition.

3.1. Tokenization, Architecture, and Output

We begin by addressing the three key modeling decisions outlined in the previous section—input representation (tokenization), architecture, and output—in the context of the Hebrew language.

Recall that, as discussed in the introduction, Hebrew is an MRL with the following important characteristics: (i) grammatical relations in Hebrew are expressed via the addition of affixes; (ii) Hebrew sentences are nearly order-free; (iii) many Hebrew words have multiple meanings, which change depending on context; and (iv) Hebrew contains vocalization diacritics that are missing in nonformal scripts, implying that words that are pronounced differently can be written in the same way.

Bearing these features in mind, we first address the *last* two questions, of architectural choice and model output. As discussed in previous sections, BERT has been shown to outperform alternative architectures in sentiment analysis tasks (Radford et al. 2019). Moreover, the literature offers evidence that BERT networks effectively capture linguistic information and phrase-level information (Jawahar et al. 2019), a necessary requirement for MRLs (Tsarfaty et al. 2020). Accordingly, we decided to use BERT as our base model, with the default architecture. For the output task, we used BERT's default fill-in-the-blank task. The fill-in-the-blank task has the advantage of understanding bidirectional context, which corresponds to the order-free property of Hebrew sentences.[3]

With respect to the *input*—the granularity of the tokens—the literature on MRLs, and Hebrew specifically, is inconclusive. Belinkov et al. (2017) and Vania et al. (2018) showed that character-based representation, which is becoming increasingly popular, is better than word-based representation for learning Hebrew morphology, especially for low-frequency words. For sentiment tasks, however, Amram et al. (2018) and Tsarfaty et al. (2020) showed that a word-based representation yields better predictions than a char-based representation. With regard to subword representations, Klein and Tsarfaty (2020) suggested (but did not verify) that, for BERT for Hebrew, morpheme-based subwords are likely to be preferable to *n*-gram-based subwords. A similar argument was made for Arabic, which is the closest MRL language to Hebrew (Antoun et al. 2020).

To understand what causes differences in findings among different researchers, consider the following three examples:

1. First consider the word *na'al*. na'al can be translated as either "locked" (e.g., he *locked* the door), "a shoe," or the past, singular tense of the verb *wearing* (a shoe). It is also often used as a slang term for "stupid." The actual semantic meaning of na'al in a sentence is derived from the context. In that respect, a *high-level text granularity* (such as a word-based representation) might be the preferable choice for representing Hebrew, as it is better in capturing semantic meanings in context (Pota et al. 2019).

2. The next word is *na'alo*, which is an inflection of the word *na'al* with the suffix "O." na'alo can refer to either "his shoe" or "locked it." In that respect, a *finer text granularity*, such as char-based, which is better at learning morphology, might be preferred.

3. Finally, consider the splitting of the word *na'alo*. Here, a meaningful splitting would be *na'al-o*. However, such a splitting can be achieved only with *morpheme-based subwords*, using a tool such as YAP (Yet Another Parser, by More et al. 2019). The alternative,

n-gram-based subwords will result in additional splitting, which might have lower semantic meaning than morpheme-based subwords, yet higher robustness to OOV.

Given the above discussion, we hypothesize that subword representations (*n*-gram- or morpheme-based representations), which balance semantic meaning with morphology, will best capture the features of the Hebrew language and will yield better performance for various language tasks, compared with character-based and word-based representations. Comparing *n*-gram-based subwords with morpheme-based subwords, we expect the latter to have an advantage on token-level tasks that require a good "understanding" of the language features; yet, a morpheme-based representation might not have such an advantage in document-level downstream tasks.

To examine our hypotheses, we first train and evaluate multiple small-size BERT models that differ by the granularity of the input. We then choose the best-performing architecture and retrain the model on a much larger corpus.

3.2. Comparison Analysis of Tokenization Approaches

We examine five alternative text representations: one char-based representation; two *n*-gram-based subword representations, which differ in total vocabulary size (30,000 tokens versus 50,000 tokens); a morpheme-based subword representation; and a word-based representation, which considers all words in the corpus, after trimming terms in the lowest fifth quantile according to their term frequency (vocabulary size of over 53,000 tokens).

To compare between the input alternatives, we first train small-sized base BERTs on a Hebrew Wikipedia dump.[4] Our working assumption is that the performance of a small-sized BERT is monotonic with the model's performance when trained on a larger corpus with the same parameters, yet requires significantly fewer resources.[5]

We evaluate the models' performance on two common unsupervised language tasks and on three downstream tasks:

1. The unsupervised language tasks are the following:

a. Fill in the blank—the ability to fill in a missing token, tested on a newspaper article[6] and a fairy-tale data set.[7] Performance was measured with sequence perplexity ($PP(W)$), a common measure to examine the ability of a language model to evaluate the correctness of sentences in a sample set. Perplexity of a sequence W with N tokens ($W = \{w_1, w_2, \ldots, w_n\}$) is calculated as the exponential average log-likelihood of the sequence ($PP(W) = exp\{-\frac{1}{N}\sum_i^N log_{p_\theta}(w_i|w_{<i})\}$, where $log_{p_\theta}(w_i \mid w_{<i})$ is the log-likelihood of the ith token conditioned on the preceding tokens, according to the language model).

b. Generalizability to OOV—the ability of the language model to generalize beyond the trained corpus (Wikipedia vocabulary), as measured by the percentage of tokens in a testing set for which the language model could not predict the term embedding. As a testing set, we used the corpus reported in Amram et al. (2018).

2. The downstream classification tasks are the following:

a. Named-entity recognition—the ability of the model to classify named entities in text, such as persons' names, organizations, and locations, tested on a labeled data set from Mordecai and Elhadad (2005), and evaluated with F_1 score ($2 \times \frac{precision \times recall}{precision + recall}$). Following Mordecai and Elhadad (2005), NER was trained for five epochs with learning rate (lr) = 5e-5, and evaluated using a fourfold cross-validation approach.

b. Part of speech—the ability of the model to classify the grammatical role that a word or phrase plays in a sentence (e.g., noun, pronoun, verb), tested on a labeled Israeli newspaper data set from Sima'an et al. (2001) and evaluated with F_1 score. To train and test POS detection, we used the code provided by Klein and Tsarfaty (2020).

c. Polarity analysis—tested on the polarity data that were collected and labeled by Amram et al. (2018) and evaluated with F_1 score. Following Amram et al. (2018), the model was trained for 10 epochs with lr = 5e-5.

The results of this comparison are shown in Table 1. To ensure fair comparison, we trained and tested all

Table 1. Task Performance Comparison for Different Input Alternatives

Metric	Fill in the blank (perplexity)	OOV (%)	NER (F_1 score)	POS (F_1 score)	Polarity analysis (F_1 score)
Chars (1,000)	1.17	~0	0.74	0.92	0.69
n-gram (30,000)	4.4	~0	0.79	0.90	0.79
n-gram (50,000)	5.7	~0	0.79	0.92	0.71
Morpheme based	8.9	0.75	0.92	0.95	0.65
Word based	209,830	50	0.86	N/A[a]	0.43

[a]POS could not be computed due to the high OOV percentage.

models on the same data sets, using the same settings and evaluation methodology across methods.

For the unsupervised tasks (fill in the blank and OOV), the subword representations (*n*-gram based and morpheme based) performed similarly well, and substantially outperformed the word-based representation. Specifically, all subword methods were able to capture OOV tokens, with the exception of special emojis. The performance for the fill-in-the-blank task is monotonic with respect to the dictionary size. Char-based representation outperformed the subword representations on the fill-in-the-blank task and performed equally well on the OOV task. This is not surprising, as smaller-sized dictionaries leave less room for mistakes.

For each of the downstream tasks, in line with our hypothesis, the top-performing tokenization was a subword representation. Specifically, *n*-gram-based tokenization performed best on the POS task, whereas morpheme-based tokenization achieved the best performance on NER. For polarity analysis, the *n*-gram-based approach with the smaller dictionary (30,000) performed significantly better than all other approaches.

These results suggest that (1) there is no single representation that is optimal for the entire set of tasks; (2) for each task, there is at least one subword representation that outperforms both the char- and word-based representations; and (3) for a sentiment analysis task, which is the focus of this work, an *n*-gram-based subword representation with a smaller dictionary yields the highest performance. On the basis of these results, we selected the latter tokenization for our model.

3.3. Final Model

In line with the specifications outlined above, we trained a large-size BERT on both the Wikipedia corpus and an OSCAR corpus (Ortiz Suárez et al. 2020), with a small-size *n*-gram-based subword dictionary. For the Hebrew language, OSCAR contains a corpus of size 9.8 GB, including 1 billion words and over 20.8 million sentences (after deduplicating the original data). The size of the training corpus for the final model was 10.5 GB.

We used a PyTorch implementation of transformers in Python (Wolf et al. 2020) to train a base-BERT network for four epochs, with a learning rate equal to 5e-5, using the Adam optimizer in batches of 128 documents each.

The performance of the final model is reported in Table 2. We compared the performance of this model with the performance of (i) the (non-BERT) models reported in Amram et al. (2018) (polarity analysis), More et al. (2019) (POS tagging task), and Bareket and Tsarfaty (2021) (NER tagging, the first model developed for NLP tasks in Hebrew and considered the state of the art), and (ii) fine-tuned mBERTs for each downstream task. We further compared HeBERT to the AlephBERT model (Seker et al. 2021), a newly developed model that is based on the architecture guidelines in this paper, yet was trained on significantly larger corpora.

The results show that although mBERT outperformed HeBERT in an unsupervised task (fill in the blank), HeBERT performed better on supervised tasks, even when compared with the current SOTA. Of note, mBERT contains only 2,000 tokens in Hebrew (compared with 30,000 in HeBERT). HeBERT's higher performance in supervised tasks is thus not surprising. Interestingly, we find that AlephBERT does not improve on HeBERT, despite being trained on larger corpora. This result is consistent with the findings of Seker et al. (2021).

4. HebEMO: A Model for Polarity Analysis and Emotion Recognition

In this section, we develop HebEMO, a model for sentiment analysis, including polarity analysis and emotion recognition. HebEMO, which is based on HeBERT, predicts sentiments at a document level. As elaborated in what follows, in our case, a *document* is a single user-generated comment on a news website. The development of the model is based on three main elements: (i) data collection, (ii) data annotation, and (iii) fine-tuning of HeBERT.

4.1. Data Collection

The data collected for this study were compiled from user comments that were posted to Israeli news websites in response to COVID-19-related articles during the first year of the COVID-19 pandemic (January to December 2020), a highly emotional period (Pedrosa et al. 2020).

Our selection of news sites was inspired by a 2016 statement by Israel's then-president Reuven (Rubi) Rivlin,

Table 2. HeBERT Performance Compared with Alternative Models

	Task				
Metric	Fill in the blank (perplexity)	OOV (%)	NER (F_1 score)	POS (F_1 score)	Polarity analysis (accuracy)
HeBERT	3.24	~0	0.96	0.955	0.94
Current SOTA	Not reported	8	0.84	0.97	0.89
mBERT	1	0	0.94	0.933	0.92
AlephBERT	1.09	0	0.97	0.956	0.94

according to which Israeli society is composed of four equally sized "tribes" that are culturally different (and hence might express emotions slightly differently). Of these, three comprise Hebrew-speaking Jews—namely, secular, national-religious, and ultra-Orthodox (Haredi)—and the fourth tribe is Israel's Arab population (Steiner 2016). Each group is represented in both politics and the media.

Accordingly, we collected data from three popular Israeli news sites that, respectively, represent the three Hebrew-speaking tribes: Ynet,[8] which is identified with the secular tribe (with a slight left-wing political leaning); *Israel Hayom*[9] (*Israel Today*), which is identified with the national-religious tribe (with a slight right-wing political leaning), and *Be-Hadre Haredim*[10] (*In Haredis' Rooms*), which represents the ultra-Orthodox group. Specifically, our data set contained all articles that these websites had published in 2020 and had tagged as being related to COVID-19.

For each article, we collected the article's text, its date of publication, the section in the news site in which it was published (e.g., news, health, sports), the author, and the comments section. We excluded from the data set comments that did not contain Hebrew words and comments with fewer than three words. We further merged repeated consecutive characters (e.g., three or more identical punctuation symbols) and removed links and double spaces. The compiled corpus, summarized in Table 3, contained over half a million comments on 10,794 titles in various sections.

4.2. Data Annotation

We annotated a total of 4,000 comments. Comments were selected for annotation following an innovative active learning approach (Li et al. 2012) to minimize the well-known imbalance problem in the emotion recognition literature (Acheampong et al. 2020). The annotation process we used is described below and illustrated in Figure 2.

Table 3. Description of the Collected Data

Source	Section	# titles	# comments
Ynet	Activism	3	80
	Article	500	30,053
	Articles	4,506	156,174
	Dating	34	793
	Digital	124	2,067
	Economy	1,486	55,521
	Entertainment	153	7,000
	Food	41	4,064
	Health	271	8,112
	Judaism	63	4,321
	News	138	8,214
	Sport	71	363
	Vacation	181	10,467
	Wheels	32	964
Israel Hayom	Article	2,651	71,372
	Opinion	36	392
Be-Hadre Haredim	News	191	805

We initialized our iterative process in Step 1 with a naive unsupervised lexicon-based approach. For this step, we Google-translated EmoLex: a freely available English-language polarity and emotion dictionary (Mohammad and Turney 2013). EmoLex contains a list of manually collected (via crowdsourcing) English words classified according to one or more of the eight basic emotions and two polarity values (positive and negative). We then used the translated dictionaries to score the entire set of *lemmatized* comments in our data set. Lemmatization was achieved with UDPipe (Straka et al. 2016).

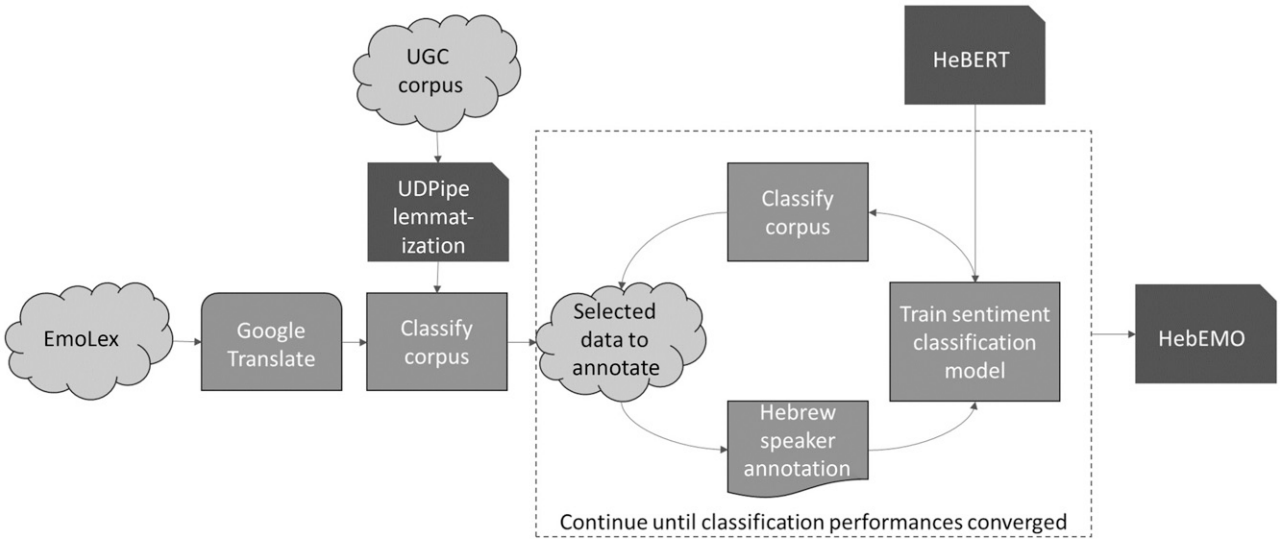
In Step 2, given the initial sentiment scores generated in Step 1, we selected a set of 150 comments, of which 75 comments had received the highest positive polarity scores, and 75 had received the highest negative polarity scores. Similarly, for each of the eight emotions, we selected a set of 75 comments in which the emotion was highly expressed, and another 75 comments in which the emotion was not expressed. The resulting set, after removing duplicate comments, comprised a total of 1,500 initially labeled comments.

We then turned to Prolific,[11] a trusted online labor and research platform, to manually reannotate the 1,500 comments. Each comment was annotated by at least three distinct native Hebrew-speaking Prolific workers. Specifically, annotators were asked to rate individual comments' polarity on a symmetric five-point scale (strongly negative, negative, neutral, positive, and strongly positive), and to rate the expression of each emotion in the comment on a polar three-point scale (not expressed (in the comment), expressed, strongly expressed). The participants were given the context of the comment (i.e., the title of the news article on which the comment was posted). Each participant annotated 20 randomly selected comments.

The reliability levels of workers' annotations were then computed with Krippendorff's (1970) alpha, a measure of interrater agreement. We measured reliability independently for each sentiment in a comment, using coarser sentiment scales of polarity (positive, neutral, and negative) and emotion (expressed or not expressed). For example, if two raters, i and j, rated the emotion anger in a comment c as $L^i_{c,anger}$ = *expressed* and $L^j_{c,anger}$ = *strongly expressed*, we computed their mutual response as "agreement" (formally, the observed agreement between the raters was $\delta(L^i_{c,anger}, L^j_{c,anger}) = 0$). If the ratings were $L^i_{c,anger}$ = *expressed* (or *strongly expressed*) and $L^j_{c,anger}$ = *not expressed*, we computed the raters' mutual response as "disagreement" ($\delta(L^i_{c,anger}, L^j_{c,anger}) = 1$). We then excluded comments for which the Krippendorff's alpha of the sentiment annotation was lower than 0.75.

In Step 3, we trained an initial HeBERT-based sentiment (supervised) classifier (see details in Section 4.3)

Figure 2. Iterative Annotation Process



on the crowd-annotated data and predicted polarity and emotion scores for the remainder of the corpus. We then repeated Steps 2 and 3 until the performance of our classifier converged. Convergence occurred after three iterations and a total of 4,000 partially labeled comments (partially means that the raters agreed on at least one sentiment). Tables 4 and 5 summarize the number of comments for each sentiment (polarity and emotion, respectively) for which there was high agreement among raters and the percentage of the comments that express this sentiment. For example, the expression/nonexpression of the emotion anger was labelled in 1,979 distinct comments; among these, anger was expressed in 78% of the comments and not expressed in 22% of the comments.

Interestingly, though we attempted to balance the expression and nonexpression of each sentiment in our labelled data, our raters had significantly lower agreement on positive sentiments—specifically, positive polarity, expression of happiness, surprise, and trust, and nonexpression of anger and disgust—than on negative sentiments. In line with the theory of Plutchik (1980), we observed high negative correlation between emotions that are located opposite each other in Plutchik's wheel of emotion, and positive correlation between closely related emotions (see Table 6).

The final classification model was named HebEMO.

Table 4. Summary of the Polarity Data

Polarity	# labeled comments	% comments (%)
Positive	253	13.8
Neutral	55	3
Negative	1,525	83.2

4.3. Fine-Tuning of HeBERT: The Classification Model

We modeled our classification algorithm by fine-tuning HeBERT for a document-level classification task. Prediction probabilities were computed with a linear activation function, applied on the dropout layers of BERT's ouput:

$$Linear_{in=768,\ out=2}(Dropout_{p=0.1}(\textbf{HeBERT Outputs})).$$

We treated the polarity task as a multinomial problem with three classes (positive, neutral, and negative); emotions were modeled as independent dichotomous classification tasks (expressed and not expressed), as multiple emotions can coexist in a single comment. Attempts to merge emotion pairs (e.g., joy–sadness) into a single classification category yielded lower performance.

To train and evaluate our model, we randomly partitioned the corpus into training (70%), validation (15%), and test (15%) sets. In order to avoid data leakage, the tokenization process (in HeBERT) was not trained on the UGC data set. We repeated the training and evaluation process following a bootstrap approach with five

Table 5. Summary of the Emotion Data

Emotion	# labeled comments	% comments (%)
Anger	1,979	78
Disgust	2,115	83
Anticipation	681	58
Fear	1,041	45
Joy	2,342	12
Sadness	998	59
Surprise	698	17
Trust	1,956	11

Table 6. Pearson Score for Correlation Among Emotions Identified by Human Raters

	Anger	Disgust	Anticipation	Fear	Joy	Sadness	Surprise	Trust	Polarity
Anger	1.00								
Disgust	0.46	1.00							
Anticipation	0.10	0.09	1.00						
Fear	0.15	0.11	0.14	1.00					
Joy	0.25	0.27	0.12	0.11	1.00				
Sadness	0.21	0.16	0.13	0.28	0.12	1.00			
Surprise	0.06	0.04	0.10	0.15	0.05	0.12	1.00		
Trust	0.27	0.31	0.11	0.07	0.41	0.08	0.07	1.00	
Polarity	0.47	0.44	0.11	0.09	0.36	0.14	0.05	0.40	1.00

samples (each generated a different data partition) and examined the stability of our results.

5. Results

We applied HebEMO to our annotated data set and examined its performance, as measured by precision, recall, F_1 score, and overall accuracy of the expressed sentiment. Table 7 presents the performance of our model on the polarity task, and Table 8 presents the performance for emotion recognition. The weighted average performance across all sentiments is $F_1 = 0.931$, with overall accuracy of 0.91. With the exception of the emotion surprise, the performance of the model ranges between F_1 scores and accuracy of 0.80–0.96. These performance levels, as far as we know, exceed those of state-of-the-art English-language models for UGC emotion recognition (Mohammad et al. 2018, Ghanbari-Adivi and Mosleh 2019).

The emotion surprise is known to be hard to detect. As mentioned in Zhou et al. (2020), the best reported F_1 score for this emotion in English was found to be as low as 0.19 (Mohammad et al. 2018). In our data set, the amount of labeled data for surprise, as well as for its opposing counterpart on the wheel of emotion, anticipation (Plutchik 1980), was also the lowest among all emotions (see Table 5), implying that detecting these two emotions constitutes a challenging labeling task even for human annotators.

Next, we retrained HebEMO on the polarity data reported by Amram et al. (2018). Amram et al. (2018) collected comments that were written in response to official tweets posted by the Israeli president, Mr. Reuven Rivlin, between June and August 2014 (a total of 12,804 Hebrew comments). The authors manually annotated the comments with the labels *supportive* (positive), *criticizing* (negative), and *off-topic* (neutral), and published a partitioned data set (training and validation) for the benefit of comparisons between language models.

The performance of our model is presented in Table 9, along with the improvement/deterioration in performance as compared with the SOTA model reported in Amram et al. (2018). The results show that in most aspects, with the exception of off-topic precision, our model's performance exceeds that of the SOTA model. The improvement is significant at the 95% confidence level.

Table 7. HebEMO Performance on the Polarity Task in the UGC Data

	Precision	Recall	F_1 score
Positive	0.96	0.92	0.94
Neutral	0.83	0.56	0.67
Negative	0.97	0.99	0.98
Accuracy			0.97

5.1. Robustness and Additional Comparisons

In this section, we examine the robustness of HebEMO to our data annotation and selection for training methodologies. (Recall that our data set is balanced on each of the emotions in the data and contains only comments with high agreement on their emotions.) For our robustness analyses, we collected and annotated two additional data sets. The first contains a random set of comments taken from our in-domain data set (that is, comments that were posted in response to COVID-related news articles). The second is a random set of comments taken from an out-of-domain data set containing comments that were posted in response to non-COVID-related articles from the same news sites. We then labeled each emotion in each comment according to the majority vote of the annotators. These labels served as an alternative "ground truth"

Table 8. HebEMO Performance on the Emotion Detection Task in the UGC Data

Emotion	F_1	Precision	Recall	Accuracy
Anger	0.93	0.93	0.93	0.96
Disgust	0.90	0.90	0.91	0.95
Anticipation	0.82	0.82	0.82	0.82
Fear	0.82	0.84	0.82	0.82
Joy	0.95	0.96	0.93	0.98
Sadness	0.81	0.82	0.80	0.83
Surprise	0.62	0.62	0.62	0.74
Trust	0.91	0.90	0.92	0.96

Table 9. The Performance of HebEMO When Trained on the Polarity Corpus Reported by Amram et al. (2018)

	Precision	Recall	F_1 score
Positive	0.94 (+0.02)	0.97 (+0.02)	0.95 (+0.01)
Negative	0.91 (+0.07)	0.88 (+0.01)	0.90 (+0.05)
Off-topic	0.69 (−0.31)	0.51 (+0.50)	0.59 (0.00)
Accuracy			0.93 (+0.03)

indicator of each emotion, one that was likely to be less strict compared with interrater agreement.

Tables 10 and 11 present the performance of the model on random in-domain comments and on out-of-domain comments, respectively. Each table shows the performance both for data labeled by majority vote and for data labeled according to interrater agreement. The results show that though, as expected, the performance of the model on these new data sets is poorer than its performance on in-sample data, it nevertheless performs relatively well on both in-domain and out-of-domain data, even in the presence of low-agreement comments (average F_1 score of 0.7). Interestingly, the performance of the model on out-of-sample (in-domain) comments is statistically equal to its performance on out-of-domain comments.

We further compared our HebEMO model to a comprehensive set of alternative approaches, including alternative BERT models (specifically, mBERT and AlephBERT), and different settings of ML-based classifiers. The results, detailed and presented in Online Appendix B, show that (1) HebEMO that uses HeBERT yields statistically the same prediction performance as a model that uses AlephBERT, and (2) both these models are superior to all other alternatives.

6. Summary and Implications for Decision Makers

This paper presented two new tools that contribute to the development of Hebrew-language sentiment analysis capabilities: (i) HeBERT, the first Hebrew BERT model and a new state-of-the-art model for multiple Hebrew NLP tasks, and (ii) HebEMO, a tool for polarity analysis and emotion recognition from Hebrew UGC. The tools presented in this paper offer a great advantage for decision makers, as well as a clear contribution to the data science literature.

On the managerial side, understanding consumers' emotions with a high degree of accuracy translates into a great economic advantage. Numerous studies have established the impact of consumer satisfaction and purchase experience on willingness to pay and on the probability of being a repeat customer (Woodruff 1997, Chitturi et al. 2007, Meyer and Schwager 2007). Accordingly, an accurate sentiment analysis model such as HebEMO can provide marketers with substantial value, in enabling them to continuously monitor consumers' emotions, to identify points of failure, and to better target consumers according to their experience.

Methodologically, although HeBERT was developed for the purpose of optimizing sentiment analysis, we showed that it outperforms mBERT in a variety of supervised language tasks. This finding is consistent with the literature that proposes that language-specific models are better than multilingual models (Antoun et al. 2020). HeBERT also showed better performance than the current (non-BERT) SOTA Hebrew-language model. We further note that AlephBERT, which is based on the architecture guidelines provided herein (in Chriqui and Yahav 2021) but is trained on a larger corpus, does not improve upon HeBERT's performance in the various tasks.

Moreover, for the task of extracting sentiments from UGC, we showed that the high-complexity morpheme-based model, which aims to "understand" features of the language, performed less well than a parsimonious model that did not address the language features (*n*-gram-based subwords). For the latter input representation, a smaller-size dictionary was better than a larger-size dictionary. A plausible explanation for these results is the well-known *overfitting* problem in ML: a model that is too closely fitted to terms in a language

Table 10. HebEMO Performance Against Random Comments from the Corpus

	Label by majority decision			Label by interrater agreement		
Emotion	F_1 score	Recall	Precision	F_1 score	Recall	Precision
Anger	0.70	0.72	0.70	0.84	0.90	0.81
Disgust	0.70	0.68	0.73	0.87	0.90	0.84
Anticipation	0.61	0.65	0.62	0.87	0.89	0.88
Fear	0.66	0.66	0.68	0.89	0.91	0.88
Joy	0.75	0.74	0.75	0.80	0.90	0.75
Sadness	0.67	0.67	0.68	0.82	0.80	0.87
Surprise	0.67	0.67	0.68	1.00	1.00	1.00
Trust	0.70	0.74	0.67	0.78	0.90	0.72

Table 11. HebEMO Performance Against Random Out-of-Domain Comments

	Label by majority decision			Label by interrater agreement		
Emotion	F_1 score	Recall	Precision	F_1 score	Recall	Precision
Anger	0.70	0.70	0.70	0.81	0.82	0.80
Disgust	0.67	0.66	0.69	0.78	0.77	0.80
Anticipation	0.62	0.63	0.62	0.66	0.69	0.68
Fear	0.58	0.57	0.64	0.68	0.63	0.96
Joy	0.78	0.78	0.79	0.89	0.89	0.89
Sadness	0.68	0.68	0.68	0.82	0.80	0.91
Surprise	0.62	0.60	0.64	0.49	0.50	0.49
Trust	0.70	0.70	0.69	0.75	0.74	0.76

might overlook phenomena that are "hidden" in the text, or that exist at a higher level of abstraction, such as polarity and emotions.

In future works, we will examine additional training tasks for HeBERT, including NSP and PMI masking, to understand how they influence HeBERT's performance on downstream tasks.

Endnotes

[1] The data for this paper were collected according to accepted ethical standards.

[2] See https://github.com/avichaychriqui/HeBERT.

[3] Given the recent concerns raised regarding the contribution of NSP to model performance on various NLP tasks, we decided to refrain from using this task. Rather, following Liu et al. (2019b), the masked training text input of HeBERT was blocks of text from single documents, rather than single sentences.

[4] This was done in September 2013 (retrieved from https://u.cs.biu.ac.il/yogo/hebwiki/). The data set includes over 63 million words and 3.8 million sentences (total size, 650 MB).

[5] We acknowledge that the relative performance of the models on the small-size data sets can serve only as a proxy for their relative performance when trained on larger corpora. Nevertheless, because of the complexity and cost of training multiple full-size models, we compare the models based solely on the Wikipedia corpus.

[6] See https://www.haaretz.co.il.

[7] See https://benyehuda.org/, a volunteer-based free digital library expanding access to Hebrew literature.

[8] See https://www.ynet.co.il/.

[9] See https://www.israelhayom.co.il/.

[10] See https://www.bhol.co.il/.

[11] See https://www.prolific.co/.

References

Abdul-Mageed M, Diab M, Korayem M (2011) Subjectivity and sentiment analysis of modern standard Arabic. *Proc. 49th Annual Meeting Assoc. Comput. Linguistics: Human Language Tech.* (Association for Computational Linguistics, Stroudsburg, PA), 587–591.

Acheampong FA, Wenyu C, Nunoo-Mensah H (2020) Text-based emotion detection: Advances, challenges, and opportunities. *Engineering Reports* 2(7):e12189.

Adamopoulos P, Ghose A, Todri V (2018) The impact of user personality traits on word of mouth: Text-mining social media platforms. *Inform. Systems Res.* 29(3):612–640.

Ahmad Z, Jindal R, Ekbal A, Bhattachharyya P (2020) Borrow from rich cousin: transfer learning for emotion detection using cross lingual embedding. *Expert Systems Appl.* 139:112851.

Ahorsu DK, Lin CY, Imani V, Saffari M, Griffiths MD, Pakpour AH (2020) The fear of COVID-19 scale: development and initial validation. *Internat. J. Mental Health Addiction*, ePub ahead of print March 27.

Amram A, David AB, Tsarfaty R (2018) Representations and architectures in neural sentiment analysis for morphologically rich languages: A case study from modern Hebrew. *Proc. 27th Internat. Conf. Comput. Linguistics* (Association for Computational Linguistics, Stroudsburg, PA), 2242–2252.

Antoun W, Baly F, Hajj H (2020) AraBERT: Transformer-based model for Arabic language understanding. *Proc. 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection* (European Language Resource Association, Marseille, France), 9–15.

Argaman O (2010) Linguistic markers and emotional intensity. *J. Psycholinguistic Res.* 39(2):89–99.

Bareket D, Tsarfaty R (2021) Neural modeling for named entities and morphology (NEMO^2). *Transactions of the Association for Computational Linguistics* (MIT Press, Cambridge, MA), 9:909–928.

Belinkov Y, Durrani N, Dalvi F, Sajjad H, Glass J (2017) What do neural machine translation models learn about morphology? Preprint, submitted April 11, https://arxiv.org/abs/1704.03471.

Bellstam G, Bhagat S, Cookson JA (2020) A text-based analysis of corporate innovation. *Management Sci.* 67(7):4004–4031.

Bojanowski P, Joulin A, Mikolov T (2015) Alternative structures for character-level RNNs. Preprint, submitted November 19, https://arxiv.org/abs/1511.06303.

Chatterjee A, Narahari KN, Joshi M, Agrawal P (2019) SemEval-2019 task 3: EmoContext contextual emotion detection in text. *Proc. 13th Internat. Workshop Semantic Eval.* (Association for Computational Linguistics, Stroudsburg, PA), 39–48.

Chitturi R, Raghunathan R, Mahajan V (2007) Form vs. function: How the intensities of specific emotions evoked in functional vs. hedonic trade-offs mediate product preferences. *J. Marketing Res.* 44(4):702–714.

Chriqui, A, Yahav I (2021). HeBERT & HebEMO: A Hebrew BERT model and a tool for polarity analysis and emotion recognition. Preprint, submitted February 3, https://arxiv.org/abs/2102.01909.

Desmet B, Hoste V (2013) Emotion detection in suicide notes. *Expert Systems Appl.* 40(16):6351–6358.

Devlin J, Chang MW, Lee K, Toutanova K (2018) BERT: Pre-training of deep bidirectional transformers for language understanding. Preprint, submitted October 11, https://arxiv.org/abs/1810.04805.

Dong L, Wei F, Tan C, Tang D, Zhou M, Xu K (2014) Adaptive recursive neural network for target-dependent Twitter sentiment classification. *Proc. 52nd Annual Meeting Assoc. Comput.*

Linguistics, vol. 2 (Association for Computational Linguistics, Stroudsburg, PA), 49–54.

Ekman P (1999) Basic emotions. Dalgleish T, Power MJ, eds. *Handbook of Cognition and Emotion* (Wiley, Hoboken, NJ), 45–60.

El-Din DM (2016) Enhancement bag-of-words model for solving the challenges of sentiment analysis. *J. Adv. Comput. Sci. Appl.* 7(1): 244–252.

Fattah K, Fierke KM (2009) A clash of emotions: The politics of humiliation and political violence in the middle east. *Eur. J. Internat. Relations* 15(1):67–93.

Fedus W, Goodfellow I, Dai AM (2018) MaskGAN: Better text generation via filling in the _. Preprint, submitted January 23, https://arxiv.org/abs/1801.07736.

Ghanbari-Adivi F, Mosleh M (2019) Text emotion detection in social networks using a novel ensemble classifier based on Parzen tree estimator (TPE). *Neural Comput. Appl.* 31(12):8971–8983.

Hemmatian F, Sohrabi MK (2019) A survey on classification techniques for opinion mining and sentiment analysis. *Artificial Intelligence Rev.* 52:1495–1545.

Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput.* 9(8):1735–1780.

Hogenboom A, Bal D, Frasincar F, Bal M, de Jong F, Kaymak U (2013) Exploiting emoticons in sentiment analysis. *Proc. 28th Annual ACM Sympos. Appl. Comput.* (Association for Computing Machinery, New York), 703–710.

Jawahar G, Sagot B, Seddah D (2019) What does BERT learn about the structure of language? *Proc. 57th Annual Meeting Assoc. Comput. Linguistics* (Association for Computational Linguistics, Stroudsburg, PA), 3651–3657.

Joulin A, Grave E, Bojanowski P, Douze M, Jégou H, Mikolov T (2016) FastText.zip: Compressing text classification models. Preprint, submitted December 12, https://arxiv.org/abs/1612.03651.

Khan FH, Qamar U, Bashir S (2016) eSAP: A decision support framework for enhanced sentiment analysis and polarity classification. *Inform. Sci.* 367:862–873.

Kim Y (2014) Convolutional neural networks for sentence classification. Preprint, submitted August 25, https://arxiv.org/abs/1408.5882.

Kim-Prieto C, Diener E (2009) Religion as a source of variation in the experience of positive and negative emotions. *J. Posit. Psychol.* 4(6):447–460.

Klein S, Tsarfaty R (2020) Getting the## life out of living: How adequate are word-pieces for modelling complex morphology? *Proc. 17th SIGMORPHON Workshop Comput. Res. Phonetics, Phonology, Morphology* (Association for Computational Linguistics, Stroudsburg, PA), 204–209.

Kövecses Z (2003) *Metaphor and Emotion: Language, Culture, and Body in Human Feeling* (Cambridge University Press, Cambridge, MA).

Kratzwald B, Ilić S, Kraus M, Feuerriegel S, Prendinger H (2018) Deep learning for affective computing: Text-based emotion recognition in decision support. *Decision Support Systems* 115:24–35.

Krippendorff K (1970) Estimating the reliability, systematic error and random error of interval data. *Ed. Psych. Measurement* 30(1): 61–70.

Levine Y, Lenz B, Lieber O, Abend O, Leyton-Brown K, Tennenholtz M, Shoham Y (2020) PMI-masking: Principled masking of correlated spans. Preprint, submitted October 5, https://arxiv.org/abs/2010.01825.

Li S, Ju S, Zhou G, Lin X (2012) Active learning for imbalanced sentiment classification. *Proc. 2012 Joint Conf. Empirical Methods Natural Language Processing Comput. Natural Language Learn.* (Association for Computational Linguistics, Stroudsburg, PA), 139–148.

Liu B (2012) *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies, vol. 5 (Morgan and Claypool, San Rafael, CA).

Liu B, Zhang L (2012) A survey of opinion mining and sentiment analysis. Aggarwal C, Zhai C, eds. *Mining Text Data* (Springer, Boston), 415–463.

Liu B, Blasch E, Chen Y, Shen D, Chen G (2013) Scalable sentiment classification for big data analysis using naive Bayes classifier. *Proc. IEEE Internat. Conf. Big Data* (Institute of Electrical and Electronics Engineers, Piscataway, NJ), 99–104.

Liu R, Shi Y, Ji C, Jia M (2019a) A survey of sentiment analysis based on transfer learning. *IEEE Access* 7:85401–85412.

Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V (2019b) RoBERTa: A robustly optimized BERT pretraining approach. Preprint, submitted July 26, https://arxiv.org/abs/1907.11692.

Luo F, Li C, Cao Z (2016) Affective-feature-based sentiment analysis using SVM classifier. *Proc. IEEE 20th Internat. Conf. Comput. Supported Cooperative Work Design* (Institute of Electrical and Electronics Engineers, Piscataway, NJ), 276–281.

Medhat W, Hassan A, Korashy H (2014) Sentiment analysis algorithms and applications: A survey. *Ain Shams Engrg. J.* 5(4): 1093–1113.

Meyer C, Schwager A (2007) *Understanding Customer Experience* (Harvard Business Publishing, Boston).

Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. Preprint, submitted January 16, https://arxiv.org/abs/1301.3781.

Mohammad S, Bravo-Marquez F, Salameh M, Kiritchenko S (2018) SemEval-2018 task 1: Affect in tweets. *Proc. 12th Internat. Workshop Semantic Eval.* (Association for Computational Linguistics, Stroudsburg, PA), 1–17.

Mohammad SM, Turney PD (2013) Crowdsourcing a word-emotion association lexicon. *Comput. Intelligence* 29(3):436–465.

Mordecai NB, Elhadad M (2005) Hebrew named entity recognition. Preprint, submitted September, https://www.cs.bgu.ac.il/~elhadad/nlpproj/naama/HebNER.pdf.

More A, Seker A, Basmova V, Tsarfaty R (2019) Joint transition-based models for morpho-syntactic parsing: Parsing strategies for MRLs and a case study from modern Hebrew. Lee L, Johnson M, Roark B, Nenkova A, eds. *Transactions of the Association for Computational Linguistics*, vol. 7 (MIT Press, Cambridge, MA), 33–48.

Mudinas A, Zhang D, Levene M (2012) Combining lexicon and learning based approaches for concept-level sentiment analysis. *Proc. First Internat. Workshop Issues Sentiment Discovery Opinion Mining* (Association for Computing Machinery, New York), 1–8.

Mughaz D, Fuchs T, Bouhnik D (2018) Automatic opinion extraction from short Hebrew texts using machine learning techniques. *Computación Sistemas* 22(4):1347–1357.

Ortiz Suárez PJ, Romary L, Sagot B (2020) A monolingual approach to contextualized word embeddings for mid-resource languages. *Proc. 58th Annual Meeting Assoc. Comput. Linguistics* (Association for Computational Linguistics, Stroudsburg, PA), 1703–1714.

Ortony A, Clore GL, Foss MA (1987) The referential structure of the affective lexicon. *Cognitive Sci.* 11(3):341–364.

Pan SJ, Yang Q (2009) A survey on transfer learning. *IEEE Trans. Knowledge Data Engrg.* 22(10):1345–1359.

Patwa P, Aguilar G, Kar S, Pandey S, Pykl S, Gambäck B, Chakraborty T, Solorio T, Das A (2020) SemEval-2020 task 9: Overview of sentiment analysis of code-mixed tweets. *Proc. Fourteenth Workshop on Semantic Evaluation*, 774–790.

Pedrosa AL, Bitencourt L, Fróes ACF, Cazumbá MLB, Campos RGB, de Brito SBCS, Simões E Silva AC (2020) Emotional, behavioral, and psychological impact of the COVID-19 pandemic. *Frontiers Psych.* 11:566212.

Pennebaker JW, Francis ME, Booth RJ (2001) *Linguistic Inquiry and Word Count: LIWC2001* (Lawrence Erlbaum Associates, Mahwah, NJ).

Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. Preprint, submitted February 15, https://arxiv.org/abs/1802.05365.

Pfefferbaum B, North CS (2020) Mental health and the COVID-19 pandemic. *New England J. Medicine* 383(6):510–512.

Plutchik R (1980) A general psychoevolutionary theory of emotion. Plutchik R, Kellerman H, eds. *Theories of Emotion* (Elsevier, Amsterdam), 3–33.

Pota M, Marulli F, Esposito M, De Pietro G, Fujita H (2019) Multilingual POS tagging by a composite deep architecture based on character-level features and on-the-fly enriched word embeddings. *Knowledge-Based Systems* 164:309–323.

Radford A, Narasimhan K, Salimans T, Sutskever I (2018) Improving language understanding by generative pre-training. *OpenAI Blog* (Jule 11), https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I (2019) Language models are unsupervised multitask learners. *OpenAI Blog* (February 14), https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Ren Y, Wang R, Ji D (2016) A topic-enhanced word embedding for Twitter sentiment classification. *Inform. Sci.* 369:188–198.

Rosaldo MZ, Shweder RA, LeVine RA (1984) *Culture Theory: Essays on Mind, Self, and Emotion* (Cambridge University Press, Cambridge, MA).

Schuster M, Nakajima K (2012) Japanese and Korean voice search. *Proc. IEEE Internat. Conf. Acoustics, Speech Signal Processing* (Institute of Electrical and Electronics Engineers, Piscataway, NJ), 5149–5152.

Seker A, Bandel E, Bareket D, Brusilovsky I, Greenfeld RS, Tsarfaty R (2021) AlephBERT: A Hebrew large pre-trained language model to start-off your Hebrew NLP application with. Preprint, submitted April 8, https://arxiv.org/abs/2104.04052.

Shapira N, Lazarus G, Goldberg Y, Gilboa-Schechtman E, Tuval-Mashiach R, Juravski D, Atzil-Slonim D (2020) Using computerized text analysis to examine associations between linguistic features and clients' distress during psychotherapy. *J. Counseling Psych.* 68(1):77–87.

Sima'an K, Itai A, Winter Y, Altman A, Nativ N (2001) Building a tree-bank of modern Hebrew text. *Traitement automatique des langues* 42(2):247–380.

Steiner T (2016) President Rivlin's "four tribes" initiative: The foreign policy implications of a democratic & inclusive process to address Israel's socio-demographic transformation. Report, Institute for Policy and Strategy, Reichman University, Herzliya, Israel.

Straka M, Hajic J, Straková J (2016) UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. *Proc. 10th Internat. Conf. Language Resources Evaluation* (European Language Resources Association, Paris), 4290–4297.

Tang D, Wei F, Yang N, Zhou M, Liu T, Qin B (2014) Learning sentiment-specific word embedding for Twitter sentiment classification. *Proc. 52nd Annual Meeting Assoc. Comput. Linguistics*, vol. 1 (Association for Computational Linguistics, Stroudsburg, PA), 1555–1565.

Tay Y, Dehghani M, Bahri D, Metzler D (2020) Efficient transformers: A survey. Preprint, submitted September 14, https://arxiv.org/abs/2009.06732.

Tripathy A, Agrawal A, Rath SK (2016) Classification of sentiment reviews using *n*-gram machine learning approach. *Expert Systems Appl.* 57:117–126.

Tsarfaty R, Bareket D, Klein S, Seker A (2020) From SPMRL to NMRL: What did we learn (and unlearn) in a decade of parsing morphologically-rich languages (MRLs)? *Proc. 58th Annual Meeting of the Assoc. Comput. Linguistics* (Association for Computational Linguistics - Online), 7396–7408.

Tsarfaty R, Seddah D, Goldberg Y, Kübler S, Versley Y, Candito M, Foster J, Rehbein I, Tounsi L (2010) Statistical parsing of morphologically rich languages (SPMRL) what, how and whither. *Proc. NAACL HLT 2010 First Workshop Statist. Parsing Morphologically-Rich Languages* (Association for Computational Linguistics, Stroudsburg, PA), 1–12.

Ullah R, Amblee N, Kim W, Lee H (2016) From valence to emotions: Exploring the distribution of emotions in online product reviews. *Decision Support Systems* 81:41–53.

Vania C, Grivas A, Lopez A (2018) What do character-level models learn about morphology? The case of dependency parsing. *Proc. 2018 Conf. Empirical Methods in Natural Language Processing* (Association for Computational Linguistics, Brussels), 2573–2583.

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Guyon I, Von Luxburg U, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 30 (Curran Associates, Red Hook, NY), 5998–6008.

Wang G, Zhang Z, Sun J, Yang S, Larson CA (2015) POS-RS: A random subspace method for sentiment classification based on part-of-speech analysis. *Inform. Processing Management* 51(4):458–479.

Wierzbicka A (1994) Emotion, language, and cultural scripts. Kitayama S, Markus HR, eds. *Emotion and Culture: Empirical Studies of Mutual Influence* (American Psychological Association, Washington, DC), 133–196).

Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, et al. (2020) Transformers: State-of-the-art natural language processing. *Proc. 2020 Conf. Empirical Methods Natural Language Processing: System Demonstrations* (Association for Computational Linguistics, Stroudsburg, PA), 38–45.

Woodruff RB (1997) Customer value: The next source for competitive advantage. *J. Acad. Marketing Sci.* 25(2):139–153.

Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, et al (2016) Google's neural machine translation system: Bridging the gap between human and machine translation. Preprint, submitted September 26, https://arxiv.org/abs/1609.08144.

Yadav A, Vishwakarma DK (2020) Sentiment analysis using deep learning architectures: A review. *Artificial Intelligence Rev.* 53(6): 4335–4385.

Yamamoto Y, Kumamoto T, Nadamoto A (2014) Role of emoticons for multidimensional sentiment analysis of Twitter. Indrawan-Santiago M, Steinbauer M, Nguyen HQ, Tjoa AM, Khalil I, Anderst-Kotsis G, eds. *Proc. 16th Internat. Conf. Inform. Integration Web-Based Appl. Services* (Association for Computing Machinery, New York), 107–115.

Yin W, Kann K, Yu M, Schütze H (2017) Comparative study of CNN and RNN for natural language processing. Preprint, submitted February 7, https://arxiv.org/abs/1702.01923.

Yue L, Chen W, Li X, Zuo W, Yin M (2019) A survey of sentiment analysis in social media. *Knowledge Inform. Systems* 60: 617–663.

Zampieri M, Malmasi S, Nakov P, Rosenthal S, Farra N, Kumar R (2019) SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). *Proc. 13th International Workshop on Semantic Evaluation* (Association for Computational Linguistics, Minneapolis), 75–86.

Zhang L, Wang S, Liu B (2018) Deep learning for sentiment analysis: A survey. *WIREs Data Mining Knowledge Discovery* 8(4):e1253.

Zhong P, Wang D, Miao C (2019) Knowledge-enriched transformer for emotion detection in textual conversations. *Proc. 2019 Conf. Empirical Methods in Natural Language Processing and the 9th Internat. Joint Conf. Natural Language Processing (EMNLP-IJCNLP)* (Association for Computational Linguistics, Hong Kong), 165–176.

Zhou D, Wu S, Wang Q, Xie J, Tu Z, Li M (2020) Emotion classification by jointly learning to lexiconize and classify. *Proc. 28th Internat. Conf. Comput. Linguistics* (International Committee on Computational Linguistics, Barcelona, Spain), 3235–3245.