

A Diverse Set of Freely Available Linguistic Resources for Turkish

Duygu Altinok

Deepgram

duygu.altinok@deepgram.com

Abstract

This study presents a diverse set of freely available linguistic resources for Turkish natural language processing, including corpora, pretrained models and education material. Although Turkish is spoken by a sizeable population of over 80 million people, Turkish linguistic resources for natural language processing remain scarce. In this study, we provide corpora to allow practitioners to build their own applications and pretrained models that would assist industry researchers in creating quick prototypes. The provided corpora include named entity recognition datasets of diverse genres, including Wikipedia articles and supplement products customer reviews. In addition, crawling e-commerce and movie reviews websites, we compiled several sentiment analysis datasets of different genres. Our linguistic resources for Turkish also include pretrained spaCy language models. To the best of our knowledge, our models are the first spaCy models trained for the Turkish language. Finally, we provide various types of education material, such as video tutorials and code examples, that can support the interested audience on practicing Turkish NLP. The advantages of our linguistic resources are threefold: they are freely available, they are first of their kind, and they are easy to use in a broad range of implementations. Along with a thorough description of the resource creation process, we also explain the position of our resources in the Turkish NLP world.

1 Introduction

In recent years, with the development of transformers, natural language processing has experienced a dramatic breakthrough. Previously, learning architectures offered state-of-art solutions to many NLP tasks, such as sequence tagging and text classification. Accordingly, data-driven approaches have become the dominant technique used to process language data. This has made availability of large and high-quality language data an essential resource

for the development of NLP models. Turkish is spoken by over 80 million people, both in Turkey and across Europe, Cyprus, and Asia¹. However, despite this abundance of Turkish speakers, the number of available Turkish linguistic resources does not compare to the corresponding amount of resources available for well-studied languages such as English. Turkish is an agglutinative language with complex morphology (Göksel and Kerslake, 2005), and its morphosyntactic characteristics are challenging to handle in NLP applications. Similar challenges arise in creating linguistic resources for Turkish.

In this paper, we present a new set of Turkish linguistic resources, including corpora, pretrained spaCy models, and education material. The corpora comprise named entity recognition datasets of diverse genres, including Wikipedia articles and supplement products customer reviews. We also compiled several sentiment analysis datasets of different genres created via crawling e-commerce and movie reviews websites. The key characteristic of our corpora is their availability, as all corpora are easily accessible via their Github repos. With regard to spaCy pretrained models, to the best of our knowledge, our models are the first of their kind. Each spaCy pretrained language model includes a POS tagger, a dependency parser, a lemmatizer, a morphological analyzer, and a named entity recognizer as pipeline components. Although some web-based solutions have previously been provided, our POS taggers and dependency parsers are the first ones implemented in pure Python and are freely accessible. Our resources also include education materials. Specifically, we offer relevant information on the corpora building process, including all necessary details on web scraping, text cleaning, file formatting, and training the spaCy language models. Along with detailed tutorials about using

¹According to https://en.wikipedia.org/wiki/Turkish_language

pretrained spaCy models in Python, we also provide tutorials on Turkish linguistics, dataset formats and the general dataset compilation process. All in all, this paper presents a comprehensive collection of resources to the Turkish NLP community.

2 Background

In this section, we review available corpora and pretrained models to better contextualize the contributions of our work.

2.1 Related Turkish corpora

In a recent review of all available Turkish language resources, Çöltekin et al. (Çöltekin et al., 2022) reviews the few publicly available NER datasets. Some of these datasets, such as Yeniterzi version (Yeniterzi, 2011) of Tür et al.’s dataset (Tur et al., 2003), can be obtained through email. The aforementioned dataset includes ca. 500K words with 37,189 named entities (16,291 person, 11,715 location, 9,183 organization). Furthermore, the ITU NLP group offers three NER datasets (Şeker and Eryiğit, 2017) with the following three labels: person, organization, and location. These datasets are available from the group’s website² upon signing a licence agreement; however, the licence forbids any commercial use of the data. Another relevant dataset of 9,358 tweets has recently been presented by Eken and Tantuğ (Eken and Tantuğ, 2015); yet, its availability is unclear.

As revealed by the brief review above, currently available Turkish NER datasets are rather scarce, and their common limitations include difficulty of access, lack of commercial usability, small size, and minimal annotation. Furthermore, as concerns sentiment analysis datasets, Çöltekin et al. (Çöltekin et al., 2022) reviews only two publicly available and commercially usable datasets: one containing movie reviews and the other containing product reviews; these two datasets were introduced by Demirtaş and Pechenizkiy (Demirtaş and Pechenizkiy, 2013), respectively. Demirtaş’ movie reviews dataset, which contains 5,331 positive and 5,331 negative sentences, is scraped from a popular Turkish movie review site. Pechenizkiy’s reviews dataset is considerably smaller and contains 700 positive and 700 negative reviews scraped from an online retailer website. These datasets are available on the authors’ website. A third relevant dataset is called TREMO (Tocoglu and Alpkocak,

2018). Collected using a procedure similar to the one used to compile the ISEAR corpus (Scherer and Wallbott, 1994), TREMO is available only for non-commercial use. In summary, there are few sentiment corpora in the Turkish, and the available ones are small-sized (10K and 1.4K reviews), which is definitely not enough to train any kind of neural network-based architecture.

2.2 Turkish language processing pipelines

To date, only two NLP pipelines have been implemented – namely, Zemberek (Akin and Akin, 2007) and ITU Turkish NLP Web Service (Eryiğit, 2014). Zemberek is an open-source application written in Java for various NLP tasks such as tokenization, sentence boundary detection, morphological analysis and language identification. The other pipeline is ITU Turkish NLP Web Service which, as suggested by its name, is provided as a web service. This pipeline contains a tokenizer, sentence boundary detector, deasciifier, vowelizer, spelling corrector, Turkish text detector, morphological analyzer and disambiguator, named- entity recognizer, and dependency parser components. However, a limitation of ITU Turkish NLP Web Service is that, despite being a full pipeline, it is not easy to use in code; specifically, one needs to require an API token from ITU NLP group and curl the API with input text. Another limitation of this pipeline is that it is not open-source. As suggested by the brief review above, the situation with Turkish NLP pipelines leaves much room for improvement; for a language spoken by 80 million people, there are only two pipelines – one without syntax components such as POS tagger and dependency parser and the other not easily accessible. This is complicated by the fact that a decent performing POS tagger and a dependency parser for Turkish are hardly available.

3 Corpora

In this section, we present the corpora part of our set of resources. We start with our named entity and span corpora (Section 3.1), followed by sentiment analysis corpora (Section 3.2) and a small corpus of COVID-19 symptoms (Section 3.3).

3.1 Corpora for named entity and span recognition

This subsection introduces two corpora for named entity and span recognition that we compiled from

²<http://tools.nlp.itu.edu.tr/Datasets>

different resources. In what follows, we provide information about the collection process, corpus size, vocabulary size and tagset for each corpus.

3.1.1 Turkish Wiki NER Dataset

Our Turkish Wiki NER Dataset is a general-purpose named entity dataset. In essence, it is a re-annotation of a subset of the TWNERTC dataset (Sahin et al., 2017), which is a collection of automatically categorized and annotated sentences from Turkish and English Wikipedia for named entity recognition and text categorization. While the first version of TWNERTC contains 4 broad labels for person names, locations, organizations, and other sort of entities, its second version contains over 1,000 fine-grained labels. Since TWNERTC is an automatically annotated dataset, its label accuracy is not sufficient to be usable in industrial-level NER models. For that reason, for our Turkish Wiki NER Dataset, we manually annotated a set of 20,000 sentences from TWNERTC. The dataset has a redistribution and modification allowing licence (CC BY-SA 4.0). We annotated our dataset with the following 19 types of entities: cardinal numbers, dates, important events, important places, geographical places, human languages, famous laws’ names, locations, money amounts, nationalities or religious or political groups, ordinal numbers, organizations, percentages, person names, product names, quantities, time quantities, person titles, and works of art.

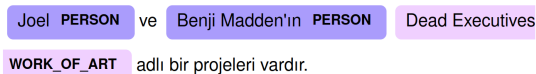


Figure 1: An example sentence from the dataset with annotated entities. The visual is created by spaCy’s visualizer displaCy (Honnibal and Montani, 2017).

The explanations and examples of labels can be found in the dataset’s Github repo.³ Our dataset contains 20,000 annotated sentences, around 357K words, with 70K of these words being unique; a total of 57,749 named-entities are labeled, and 101K words are labeled as entities. Distribution of labels in our corpus is shown in Table 1.

The data annotation was performed by our data labeling service provider.⁴ The dataset was annotated by crowd-sourcing; the labeling work

³<https://github.com/turkish-nlp-suite/Turkish-Wiki-NER-Dataset>

⁴Co-one Istanbul, <https://co-one.co/>

Tag	Count
CARDINAL	4,295
DATE	6,923
EVENT	2,392
FAC	944
GPE	10,368
LANGUAGE	822
LAW	80
LOC	1,364
MONEY	100
NORP	4,023
ORDINAL	1,711
ORG	4,583
PERCENT	182
PRODUCT	12,787
QUANTITY	990
TIME	131
TITLE	2,494
WORK_OF_ART	2,951

Table 1: Distribution of NER labels in the Turkish Wiki NER Dataset.

was done by a total of 25 annotators (15 female, 10 male). All annotators were native speakers of Turkish residing in Turkey. The dataset is available in its Github repo with CC BY-SA 4.0 licence.

3.1.2 Vitamins and Supplements NER Dataset

The Vitamins and Supplements NER Dataset is a multi-purpose NLU dataset containing customer reviews, customer review stars, as well as named entity and span annotations. User reviews were collected from a popular supplement products e-commerce website Vitaminler.com. The dataset is presented in the JSON lines format, with each instance of the dataset containing

- product name
- product’s brand name
- average star rating
- number of total ratings
- a list of customer reviews; each review consisting of a review text, review ID, star rating together with entity and span annotations.

Each customer review in the Vitamins and Supplements NER Dataset describes a customer’s experience with a supplement product in terms of

that product’s effectiveness, side effects, taste and smell, as well as comments on supplement usage frequency and dosage, active ingredients, brand, and similar products by other brands. The reviews also include pointers to customers’ health history and indications how the supplements helped in resolving customers’ health problems. As part of their health history, customers refer to certain health issues such as vitamin deficiencies, hair and skin problems, pain in several body parts (e.g., neck pain, back pain, and joint pain), as well as digestion, weight control and sleep problems. Another aspect of the collected reviews is customer demography, as customers would typically mention who they purchased the product for, including themselves or another family member (e.g., “bought the product for my baby/85-year-old mother/6-year-old daughter”), and such descriptions usually include references to gender and age. Those parts of the reviews provide valuable information about target users of the product and its effectiveness on certain demographic groups. Finally, one more valuable type of information providing meaningful clues about supplement usage habits in the population is related to who initially recommended the product to the customer (e.g., a health professional, a friend or a relative, etc.).

Ferritin eksikliğine **HEALTH_COMPLAINTS** çok iyi geldi. İleri derece **gastrit** **HEALTH_COMPLAINTS** olmasına karşın **rahatsız etmiyor** **SIDE_EFFECT** .
Kabızlık **SIDE_EFFECT** yapmıyor. **C vitamini** **BIOMOLECULE** ile emilirliliği yüksek. Depolarım doldukça **üşümem geçti ve ağrılarım azaldı** **EFFECT** .
Ağızda demir tadı ve kokusu da nerdeyse hiç yok **TASTE_SMELL** .

Figure 2: An example review from the dataset with entity and span annotations. The customer praises an iron supplement; in their reviews they explained the effects of the product as "I don't feel cold any more and my body aches relieved". Additionally the customer didn't experience any digestive system side effects despite having a digestive system condition – gastritis. The product is also said to have no metallic taste and smell. The visual is created by displaCy.

Considering the characteristics of the data, our Vitamins and Supplements NER Dataset lies at the intersection of customer review data and healthcare NLP data. Healthcare NLP datasets are conventionally compiled from a variety of genres such as doctor notes, oncology notes, radiology reports, scientific article abstracts, customer reviews for health products and contain various annotations for diag-

nosis codes, named entities, spans, and topics⁵. In view of the variety of information and annotation schemes in the healthcare domain, healthcare NLP obviously requires more than only named entity tags. In response to this need, in the Vitamins and Supplements NER Dataset, we introduced spans, which are “free” sequences of tokens. By “free” here we mean that sequence of tokens could be any sequence of tokens; that is, a sequence did not have to end/start with or contain certain POS tags (e.g., determiner, noun or verb), nor should the sequence have been a subtree in the dependency tree or provide any syntactic structure. Rather, the sequence was “free” to start and end with any token in the text, and what matters was the semantics. Since this approach blurred the concept of span boundary, there arose the question about how the annotators should label the data. In our annotation guideline, we asked the annotators to label the sequences that minimally gave the semantics of the corresponding tag, mostly leaving out “helper” words (e.g., determiners and adverbs).

To illustrate our labeling process, consider two sample user review sentences provided below in Figure 3:

Tiroid hastası olduğum için saç dökülmem var **HEALTH_COMPLAINTS** o yüzden **biotin** **BIOMOLECULE** başladım. **Saç dökülmemde gözle görülür bir azalma** **EFFECT** olduğunu söyleyebilirim.
1 buçuk aydır kullanıyorum ve **çenemin her yerinde sivilceler çıkmaya başladı** **SIDE_EFFECT**
Bir tane sivilcesi olmayan ben bu **biotin** **BIOMOLECULE** kullanmaya başladıktan 1 ay sonrası önce **çene bölgemin her yerinde, sonra alnım ve kenarları yanağımda olmak üzere sivilceler** **meydana geldi** **SIDE_EFFECT** kullanmayı hemen durdurdum. **Saç için ve tırnak için de** **yaramadı** **EFFECT** . **1 buçuk ayda yüzümü mahvetti** **SIDE_EFFECT** .

Figure 3: Two example reviews from the dataset about the same biotin product with entity and span annotations. The first review is a positive one; the customer has thyroid problems and consequent hair loss, for which the product effect is positive. The second review is a negative one; according to this customer, the product caused an acne breakout in several areas of their face including chin, cheeks and forehead. Moreover the product had no effect on overall hair and nail health. The visual is created by displaCy.

Here, labeling only a named entity, i.e. “biotin” in this case, would provide information only about active ingredients of the supplement, thus resulting in overlooking the effects and side-effects about this ingredient. A notable amount of information

⁵A list of popular healthcare NLP datasets can be found at <https://guides.lib.berkeley.edu/publichealth/healthstatistics/rawdata>

about the supplement lies in the annotated spans EFFECT and SIDE_EFFECT.

In the light of these insights from the dataset, we annotated the following 10 types of named entities: disease/symptom names, biomolecule names, the person/people who used the supplement, names of other supplement products, person/people who recommended the supplement to the customer, dosage/amount, supplement’s brand name, user demographics, ingredient substances, and other brand names mentioned in the review texts. In addition, we also annotated 4 types of spans: effects, side effects, taste and smell, and health history of the customer. The final dataset contains 2,488 instances, ca. 100K words, including 20K unique words, and around 10K entities. The distribution of named entity and span tags is summarized in Table 2.

Tag	Count
DISEASE	1,875
BIOMOLECULE	859
USER	634
OTHER_PRODUCT	543
RECOMMENDER	436
DOSAGE	471
BRAND	275
USER_DEMOGRAPHICS	192
INGREDIENT	175
OTHER_BRAND	121
EFFECT	2,562
SIDE_EFFECT	608
TASTE_SMELL	558
HEALTH_COMPLAINTS	858

Table 2: Distribution of named entity and span tags in the Vitamins and Supplements NER Dataset.

Raw data were collected by crawling Vitaminler.com. In the next step, we provided the raw data and annotation guideline to our data labeling service provider Co-one. The dataset was annotated by crowd-sourcing, and the labeling work was performed by a total of 25 annotators (15 female, 10 male). All annotators were native speakers of Turkish residing in Turkey. We also asked annotators to eliminate potentially offensive reviews and reviews containing person names (including those of the influencers). The dataset is available in its Github repo with CC BY-SA 4.0 licence.⁶ The sig-

⁶<https://github.com/turkish-nlp-suite/Vitamins-Supplements-NER-dataset>

nificance of our Vitamins and Supplements NER Dataset for the Turkish NLP world is two-fold: first, to the best of our knowledge, it is the first span recognition dataset for Turkish; second, our dataset is the first public health NLP dataset in this language.

3.2 Corpora for sentiment analysis

This subsection introduces three corpora for sentiment analysis – Beyazperde Movie Reviews Dataset (Section 3.2.1), Beyazperde Top 300 Movies Dataset (Section 3.2.2) and Vitamins and Supplements Dataset (Section 3.2.3). In what follows, we provide information about the data collection process, corpus size, and vocabulary size for each corpus.

3.2.1 Beyazperde Movie Reviews Dataset

The data for this dataset were collected by crawling popular movie reviews website Beyazperde.com. We collected URLs of 4,500 most popular movies of all times. For each movie, we crawled the movie’s name, a list of the movie’s genres, the description text, as well as the lists of directors, actors, creators, and creators of the movie’s music (i.e., composers and singers). The rating field on this website includes the number of total ratings, the number of reviews, average rating, as well as the values of the best and worst ratings on the 0-5 scale. For the reviews part, we collected all audience reviews, including review texts and review ratings. The final dataset is presented in the JSON format where each movie appears as a dictionary with general info, rating, and review information.

The final dataset contains 4,500 movies from 2,519 distinct directors. The total number of reviews is about 45K; the dataset comprises over 2.2M tokens, including 280K unique words. The star rating distribution in the review corpus is summarized in Table 3.

Star rating	Count	Star rating	Count
0.5	3,635	3.0	4,347
1.0	2,325	3.5	6,495
1.5	1,077	4.0	9,486
2.0	1,902	4.5	3,652
2.5	4,767	5.0	7,594

Table 3: Distribution of star ratings in Beyazperde Movie Reviews Dataset.

The dataset is available in its Github repo⁷ with CC BY-SA 4.0 licence.

3.2.2 Beyazperde Top 300 Movies Dataset

This dataset was also crawled from the movies website Beyazperde.com; however, this time we collected 300 top-rated movies. The data collection process and format of this dataset are identical to those of the Beyazperde Movie Reviews Dataset, with the only difference being that rating stars in the Beyazperde Top 300 Movies Dataset are highly unbalanced – namely, the numbers of 0-, 1-, 2-, and 3-star reviews are considerably lower than the corresponding numbers of 4- and 5-star ratings. Accordingly, this dataset imposes a great challenge of “finding the least/best of the best” among the best movies. The star rating distribution is shown in Table 4.

Star rating	Count	Star rating	Count
0.5	1,657	3.0	2,277
1.0	535	3.5	5,550
1.5	273	4.0	13,248
2.0	608	4.5	10,077
2.5	2,439	5.0	17,351

Table 4: Distribution of star ratings in Beyazperde Top 300 Movies Dataset.

The final dataset contains 300 top-rated movies by 218 distinct directors. The total number of reviews included in the dataset is 54K; the dataset contains over 2.4M tokens, including 50K unique tokens. Vocabulary size of this dataset is considerably smaller than that of the Beyazperde Movie Reviews Dataset. The dataset is available in its Github repo⁸ with CC BY-SA 4.0 licence.

To the best of our knowledge, both of our sentiment analysis datasets are the first of their kind, as no movie reviews of comparable size were collected before. For instance, a similar corpus published by YTU Kemik NLP Group⁹ contains reviews of mere 105 movies classified in only 3 classes (negative, positive, and neutral). Considering that modern NLP techniques require large corpora, our movie reviews datasets are sufficiently large and can thus be meaningfully used by the

⁷<https://github.com/turkish-nlp-suite/BeyazPerde-Movie-Reviews/tree/main/butun-fimler>

⁸<https://github.com/turkish-nlp-suite/BeyazPerde-Movie-Reviews/tree/main/en-iyi-fimler>

⁹<http://www.kemik.yildiz.edu.tr/>

Turkish NLP community.

3.2.3 Vitamins and Supplements Dataset

The Vitamins and Supplements NER Dataset discussed in Section 3.1.2 is a subset of a larger Vitamins and Supplements Dataset that has named entity and span annotations. The latter dataset was scraped from the supplements and health products e-commerce website Vitaminler.com. The Vitamins and Supplements Dataset includes user reviews and star ratings about supplement products. Each instance of the dataset includes a product name, brand name, average star rating value, number of customer ratings, and a list of customer reviews. A customer review includes a review text and a star rating. The dataset includes 1,052 products of 262 distinct brands with 244K customer reviews. This corpus contains 2.5M tokens, including 150K unique words. During the compilation process, we automatically eliminated potentially offensive reviews and reviews containing person names (including those of influencers). The dataset is available in its Github repo¹⁰ with CC BY-SA 4.0 licence.

3.3 Other corpora

Finally, we compiled a small corpus about COVID-19 symptoms from the popular collaborative dictionary Ekşi Sözlük,¹¹ one of the largest (over 400,000 registered users) online communities in Turkey.¹² In this community, users share information on various topics ranging from scientific subjects to everyday life issues. The data were crawled from 2 headlines – “COVID-19 Symptoms” and “Day-by-day Corona Symptoms.” This dataset, named Corona-mini, is presented in the JSON format in its Github repo.¹³

Corona-mini includes 180 instances, embracing a total of 25K tokens with 9K unique words. Each instance of the corpus is an user entry on the website. In these entries, contributors describe their experiences with common COVID-19 symptoms, including fever, cough, tiredness, muscle weakness, pain in several body parts, loss of taste and smell, insomnia, and nausea. We mined this dataset with various information extraction techniques to exhibit possible usages of new spaCy Turkish models in

¹⁰<https://github.com/turkish-nlp-suite/Vitamin-Supplements-Reviews>

¹¹<https://www.eksisozluk.com>

¹²https://tr.wikipedia.org/wiki/EkÅŖi_SÅŖzlÅk

¹³<https://github.com/turkish-nlp-suite/Corona-mini-dataset>

our video tutorial titled “Quick recipes with spaCy Turkish models”. The compilation process of this dataset was also demonstrated in our video tutorial named “How to compile NLP Datasets” (see Section 5).

4 Pretrained Models

In this section, we introduce our spaCy Turkish language models. Overall, spaCy is an industrial-strength open-source NLP library offering state-of-the-art performance with an order of magnitude exceeding other available NLP libraries (Honnibal and Montani, 2017; Honnibal, Feb 2015). spaCy also comes with a well-structured API, detailed documentation, support for issues, and an immense user community. Moreover, along with being easy to install and deploy, spaCy fits well into the Python machine learning ecosystem.

Each spaCy language model is a pipeline of pre-trained components (Honnibal, Feb 2019). Trainable components include a statistical lemmatizer, morphologizer (statistical morphological analyzer), NER, POS tagger, and dependency parser. spaCy offers “spaCy projects” for end-to-end training, packaging and sharing custom pipelines (Honnibal, Jul 2020). Accordingly, we trained our models with spaCy projects; the project template and the configuration files of each component along with the corresponding training hyperparameters are available on our Github page.¹⁴

We provide the following 3 pretrained models: `tr_core_news_trf`, `tr_core_news_lg` and `tr_core_news_md`. All these models include a vectorizer, lemmatizer, morphologizer, NER, POS tagger, and dependency parser components. The only difference among these models lies in the vectorization: while `tr_core_news_lg` and `tr_core_news_md` include static vectors, `tr_core_news_trf` is a transformer-based pipeline, meaning that word vectors for tokens are calculated and passed to the downstream components by the underlying transformer (Honnibal, Aug 2020). To train `tr_core_news_trf`, we used the dbmdz Turkish BERT model.¹⁵ `tr_core_news_lg` and `tr_core_news_md` are packaged with “static” word vectors; here, “static” means that these vectors are not learned parameters of the statistical models, and spaCy itself does not feature any algorithms

¹⁴<https://github.com/turkish-nlp-suite/turkish-spacy-models>

¹⁵<https://huggingface.co/dbmdz/bert-base-turkish-cased>

for learning word vector tables (Honnibal, Aug 2020). In order to be included in the model training, static vectors have to be separately trained and packaged.

Accordingly, we trained Floret vectors that support a vast number of subwords in a compact way (Boyd and Warmerdam, Aug 2022). For Turkish morphology, representing subwords is a critical issue. For two packages – namely, `tr_core_news_lg` and `tr_core_news_md` – we prepared two Floret vector packages: one medium-sized and the other large-sized, respectively. Training configuration of these two packages can be found in their Github repos¹⁶; the packaged vectors can be found in their Huggingface repo.¹⁷

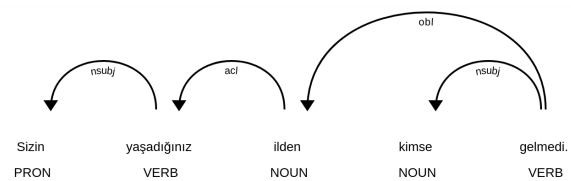


Figure 4: Dependency tree of an example sentence generated by transformer based spaCy Turkish pipeline. The sentence includes an adjectival clause “Sizin yaşadığınız” that modifies the noun “ilden”. Like adjectives, most adjectival clauses in Turkish precede the noun that they modify. The dependency relation between the clause and the modified noun is marked by *acl*. The clause has its own nominal subject “Sizin”, the relation between this token and the clause verb “yaşadığınız” is marked by *nsubj*. The clause head “ilden” functions as the oblique object of the main sentence and “kimse” is the nominal subject. All relation types come from Universal Dependencies (Nivre et al., 2020). The visual is created by displaCy.

All 3 models were trained on the same corpora: Universal Dependencies Turkish BOUN Treebank (Türk et al., 2020) was used to train the morphologizer, lemmatizer, POS tagger, and dependency parser components; the NER component was trained using our Turkish Wiki NER dataset (3.1.1) and PanX (Pan et al., 2017). We trained medium- and large- sized Floret vectors which are 300-dimensional. However, vocabulary sizes were different: while medium-sized vectors include 50K keys, large-sized vectors include 200K keys. The Floret vectors were trained on the MC4 corpus (Raffel et al., 2019). All our spaCy Turkish lan-

¹⁶[https://github.com/turkish-nlp-suite/turkish-spacy-models/tree/main/tr_vectors_web_\(lg|md\)](https://github.com/turkish-nlp-suite/turkish-spacy-models/tree/main/tr_vectors_web_(lg|md))

¹⁷[https://huggingface.co/turkish-nlp-suite/tr_vectors_web_\(lg|md\)](https://huggingface.co/turkish-nlp-suite/tr_vectors_web_(lg|md))

guage pipelines are available for download in our Huggingace repo.¹⁸

4.1 Performance and comparison

Performance of the each model on respective test-sets is shown in Table 5. Columns specify POS accuracy, morphological analysis accuracy, lemma accuracy, unlabelled attachment score for dependencies, labelled attachment score for dependencies, sentence boundary splitting F1 score, and NER F1 score. spaCy calculates sentence boundaries based on the full dependency parses. In our models, there is no pipeline component for sentence boundary detection, and spaCy library code manipulates the dependency tags during runtime to calculate the sentence boundary.

To evaluate statistical quality of our models, we compared their performance with that of pipelines for other languages, including three agglutinative languages (Hungarian (Orosz et al., 2022), Finnish, and Korean) and English (Honnibal, Feb 2019), which has a rather flat morphology. To this end, we used our best performing model, `tr_core_news_trf`, and compared it with the best performing models of the aforementioned four languages. The results of this comparison are shown in Table 6.

As revealed by the results, our pipelines are competent in statistical quality, and the corresponding values appears to be similar to those of the Finnish pipeline.

4.2 Comparison with other Turkish NLP pipelines

As discussed in Section 2, in previous research, there was only one attempt to compile an open-source, end-to-end Turkish NLP pipeline, Zemberek. In this section, we compare our spaCy Turkish pipelines to Zemberek NLP pipeline from the perspective of completeness. Zemberek pipeline does not contain any parsers for syntax, nor does it provide any pretrained NER models. Since Zemberek NLP paper was published in 2007, and the code was last updated 2 years ago, this package is outdated and does not meet the requirements of present-day NLP software. Accordingly, in this paper, we cannot make an accuracy-wise comparison (for a comparison of pipeline components, see Table 7).

Another relevant pipeline is ITU Turkish NLP Web Service which, as suggested by its name, is

provided as a web service. Although this pipeline contains both syntactic parsers and morphological analyzers, it is not easy to use in code; one needs to require an API token from the ITU NLP group and curl the API with input text; in addition, this pipeline is not open-source. Due to accessibility issues, a comparison of our spaCy Turkish models with this pipeline has to be omitted.

5 Education Material

Finally, we prepared a number of video and code tutorials to provide relevant information on the dataset collection process and dataset formats, as well as to demonstrate Python and bash scripting for cleaning and manipulating text, show possible use cases of the spaCy Turkish language models, and provide general information about Turkish linguistics. Our video tutorials, with each tutorial coming as a Youtube playlist consisting of several videos, include the following:

- How to compile NLP Datasets
- Dataset formats
- Quick recipes with spaCy Turkish models
- How to train your own spaCy language models
- All about Turkish linguistics
- Quick FAQ chatbot with semantic search and spaCy

All playlists are available on our YouTube channel¹⁹. The code tutorials are also available in our Github repo²⁰.

6 Conclusion

In this paper, we presented a diverse set of open-source linguistic resources for Turkish language processing. Our resources include corpora, pretrained spaCy language models, and education materials such as code and video tutorials. The importance of our resources for the Turkish NLP community is three-fold. The first important aspect of our resources is their accessibility. To the best of our knowledge, our Vitamins and Supplements NER Dataset is the first healthcare NLP dataset available for Turkish, while our two movie reviews datasets

¹⁸<https://huggingface.co/turkish-nlp-suite>

¹⁹<https://www.youtube.com/c/NLPwithDuygu>

²⁰<https://github.com/turkish-nlp-suite>

Model	POS acc.	Morph acc.	Lemma acc.	DEP-UAS	DEP-LAS	SENT-F	NER-F
tr_core_news_md	0.90	0.89	0.81	0.72	0.63	0.83	0.89
tr_core_news_lg	0.90	0.89	0.82	0.73	0.63	0.84	0.89
tr_core_news_trf	0.90	0.91	0.87	0.79	0.71	0.87	0.91

Table 5: Performance of spaCy Turkish models.

Model	POS acc.	Morph acc.	Lemma acc.	DEP-UAS	DEP-LAS	SENT-F	NER-F
tr_core_news_trf	0.90	0.91	0.87	0.79	0.71	0.87	0.91
hu_core_news_trf	0.97	0.94	0.98	0.91	0.87	0.99	0.91
fi_core_news_lg	0.96	0.92	0.86	0.83	0.79	0.90	0.83
ko_core_news_lg	0.95	NA	0.90	0.84	0.81	1.00	0.85
en_core_web_trf	0.98	NA	NA	0.95	0.94	0.91	0.90

Table 6: Comparison of spaCy pipelines for Turkish, Hungarian, Finnish, Korean and English.

Model	POS tagger	Dep. tagger	Lemmatizer	Morphologizer	SBD	NER
spaCy Turkish models	yes	yes	yes	yes	yes	yes
Zemberek NLP	no	no	yes	yes	yes	no

Table 7: Comparison of spaCy Turkish models with Zemberek NLP pipeline. SBD = sentence boundary detector.

are the first large-scale movie review datasets of Turkish. The second aspect of our resources is that they are the first of their kind. For instance, our spaCy Turkish language models – with a tokenizer, sentence boundary detector, vectorizer, lemmatizer, morphologizer, NER, POS tagger, and dependency parser components packaged together – are the first complete NLP pipelines for Turkish. The third important characteristic of our resources is their ease of implementation. While previous approaches have failed to provide any tools that can be easily used to solve practical text processing problems, our spaCy pipelines build on solid foundations of a multilingual and industrial-strength NLP framework. Accordingly, these pipelines are the very first easily accessible, downloadable, and industrial-strength Turkish NLP pipelines for Turkish.

Limitations

Our work reported in this paper has two limitations. First, because of the scarcity of treebanks and NER datasets for Turkish, our pretrained spaCy language models were tested on a limited amount of testsets. Second, we trained our spaCy models on general-purpose datasets compiled from Wikipedia data and formal written language resources. Accordingly, our models may not be very effective in analyzing social media texts such as Twitter data.

References

- Ahmet Afşın Akın and Mehmet Dündar Akın. 2007. Zemberek, an open source NLP framework for Turkic Languages.
- Adriane Boyd and Vincent D. Warmerdam. Aug 2022. Floret: lightweight, robust word vectors. <https://explosion.ai/blog/floret-vectors>.
- Erkin Demirtas and Mykola Pechenizkiy. 2013. *Cross-Lingual Polarity Detection with Machine Translation*. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining, WISDOM '13*, New York, NY, USA. Association for Computing Machinery.
- Beyza Eken and Ahmet Tantuğ. 2015. *Recognizing Named Entities in Turkish Tweets*. *Computer Science & Information Technology*, 5:155–162.
- Gülşen Eryiğit. 2014. *ITU Turkish NLP Web Service*. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1–4. Association for Computational Linguistics.
- A. Göksel and C. Kerslake. 2005. *Turkish: A Comprehensive Grammar*. Comprehensive grammars. Routledge.
- Matthew Honnibal. Aug 2020. Embeddings, Transformers and Transfer Learning. <https://spacy.io/usage/embeddings-transformers>.
- Matthew Honnibal. Feb 2015. Introducing spaCy. <https://explosion.ai/blog/introducing-spacy>.
- Matthew Honnibal. Feb 2019. Models & Languages. <https://spacy.io/usage/models>.

- Matthew Honnibal. Jul 2020. Projects. <https://spacy.io/usage/projects>.
- Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. [Universal Dependencies v2: An evergrowing multilingual treebank collection](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.
- György Orosz, Zsolt Szántó, Péter Berkecz, Gergo Szabó, and Richárd Farkas. 2022. [Huspacy: an industrial-strength hungarian natural language processing toolkit](#). *CoRR*, abs/2201.01956.
- Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. [Cross-lingual Name Tagging and Linking for 282 Languages](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. [Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer](#). *CoRR*, abs/1910.10683.
- H. Bahadır Sahin, Caglar Tirkaz, Eray Yildiz, Mustafa Tolga Eren, and Ozan Sonmez. 2017. [Automatically Annotated Turkish Corpus for Named Entity Recognition and Text Categorization using Large-Scale Gazetteers](#).
- Klaus R. Scherer and H G Wallbott. 1994. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66 2:310–28.
- Mansur Alp Tocoglu and Adil Alpkocak. 2018. [TREMO: A dataset for emotion analysis in Turkish](#). *Journal of Information Science*, 44(6):848–860.
- Gokhan Tur, Dilek Hakkani-Tur, and Kemal Oflazer. 2003. [A statistical information extraction system for Turkish](#). *Natural Language Engineering*, 9:181–210.
- Utku Türk, Furkan Atmaca, Şaziye Betül Özateş, Gözde Berk, Seyyit Talha Bedir, Abdullatif Köksal, Balkız Öztürk Başaran, Tunga Güngör, and Arzucan Özgür. 2020. [Resources for Turkish Dependency Parsing: Introducing the BOUN Treebank and the BoAT Annotation Tool](#).
- Reyyan Yeniterzi. 2011. [Exploiting Morphology in Turkish Named Entity Recognition System](#). In *Proceedings of the ACL 2011 Student Session*, pages 105–110, Portland, OR, USA. Association for Computational Linguistics.
- Çağrı Çöltekin, A. Seza Doğruöz, and Özlem Çetinoğlu. 2022. [Resources for Turkish Natural Language Processing: A critical survey](#).
- Gökhan Şeker and Gülşen Eryiğit. 2017. [Extending a CRF-based named entity recognition model for Turkish well formed text and user generated content](#). *Semantic Web*, 8:1–18.

ACL 2023 Responsible NLP Checklist

A For every submission:

- A1. Did you describe the limitations of your work?
Yes, I included a Limitations section just before the References section.
- A2. Did you discuss any potential risks of your work?
Not applicable. Not indeed because my paper is about building datasets and pretrained models for Turkish. There's not much of a risk to discuss.
- A3. Do the abstract and introduction summarize the paper's main claims?
Yes. I included the claims in abstract and introduction. Background work, sections 3 and 4 exhibits the evidence to my claims.
- A4. Have you used AI writing assistants when working on this paper?
Left blank.

B Did you use or create scientific artifacts?

Left blank.

- B1. Did you cite the creators of artifacts you used?
Yes, in sections 2,3,4 and 5.
- B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Yes, in section 3.
- B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Yes, sections 3 and 4.
- B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Yes, section 3. We eliminated such instances from the dataset.
- B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
yes, section 3.
- B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Yes, section 3 includes all the numbers.

C Did you run computational experiments?

Left blank.

- C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
No response.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

- C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?

No response.

- C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?

No response.

- C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?

No response.

D Did you use human annotators (e.g., crowdworkers) or research with human participants?

Yes. I worked with a commercial company for data annotations and included their name and contact information in Section 3.

- D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?

No, full annotation guideline is longer than 5 pages for both of the 2 datasets I constructed. No way would fit into this article.

- D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?

Not applicable. NA because I didn't recruit anyone, they're employees of the commercial company i worked with.

- D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?

Not applicable. NA because crowdsourcers work for a commercial company, hence this is a commercial work.

- D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?

Not applicable. NA here because my data is crawled from internet. Before crawling, we checked robots.txt of each website carefully.

- D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?

Yes, section 3.