שימוש בלמידת מכונה לצורך זיהוי שפת האם של כותב טקסט

איתמר ברץ ואיתי מונדשיין

המשימה שלנו הייתה לסווג טקסט נתון בעברית – האם נכתב על ידי דובר עברית כשפת אם או לא, עם דגש על כותבים ששפת אמם ערבית.

בשביל לבנות את המסווג שלנו השתמשנו בכלי למידת מכונה של ספריית הפייתון scikit-learn. המידע שעליו אימנו את המסווג התבסס על ערכי ויקיפדיה ופוסטים מפייסבוק בעברית, וכדוגמאות לטקסט שנכתב על ידי דובר ערבית בעברית השתמשנו בקטעים מתוך אתר חמאס בעברית וכמו כן חיבורים של תלמידים לעברית ממצרים.

השתמשנו במספר מאפיינים (features) שתוכננו כדי לתפוס הבדלים בכתיבה בין הקבוצות השונות, ביניהם משלב, תקינות תחבירית ודקדוקית, ודפוסי כתיבה טיפוסיים.

Final project in Workshop in Hebrew natural language processing

Submitted by :
Itai mondshine 207814724
Itamar Baratz 318876398

Article 1:

Background: our task is determining the native language of the author (L1) based on his writing on another language, called L2. As in the biblical story[1] (although with only peaceful uses in our mind), our goal is to classify the author's native language group using their usage of Hebrew.

The idea is identifying the language- usage patterns that are common to specific L1 language and then applying the knowledge to predict the native language of unseen texts.

So far, in the industry and in academia attempts were made to predict L1 based on texts in English (L2 = English).

We haven't been able to find previous attempts to do this task in Hebrew.

In our project, we chose to do a binary classification task – Hebrew and Arabic.
We chose this task due to several reasons:
First, Hebrew is wildly written and spoken by native Arabic speakers, so we figured that we would be able to find sources for data.

Second, we chose the Arabic language because of the similarity to the Hebrew language. Classification between two Semitic languages is a challenge we wanted to try.

1 https://he.wikipedia.org/wiki/%D7%A9%D7%99%D7%91%D7%95%D7%9C%D7%AA_(%D7%90%D7%9E%D7%A6%D7%A2%D7%99_%D7%96%D7%99%D7%94%D7%95%D7%99)#%D7%90%D7%98%D7%99%D7%9E%D7%95%D7%9C%D7%95%D7%92%D7%99%D7%94

Article 2:

The ideal data for this project are large corpora of sentences written in Hebrew by both Arabic native speakers and Hebrew native speakers.

The sources for Arabic – native texts:

1. Sentences taken from Hamas' hebrew website[2] - this corpus contains different reports written by Hamas' information department. We know that these sentences were written in Hebrew by native Arabic speakers. This corpus contains several dozen paragraphs.

2. The "Egyptian corpus" – via Facebook we made a contact with an Egyptian hebrew teacher (turns out there is a big community in Egypt that studies Hebrew) that gave us texts that were written by his students. The corpus contains dozens of paragraphs.

3. Paragraphs taken from "itztava" [3]website – this website contains different texts written by Arabic – Israeli students that study Hebrew.

The sources for Hebrew- native texts:

1. Different Wikepedia entries.
2. Different articles in Hebrew taken from news websites.

Cleaning the data

[2] /https://www.qassam.ps/hebrew

[3] /https://sites.google.com/a/etz.tzafonet.org.il/etstaba

The first problem we encountered is how to split the texts into sentences' as our model is sentence based.

we distributed the texts by ending of point/question mark/ exclamation mark. After that we removed all the sentences that had only one or two words.

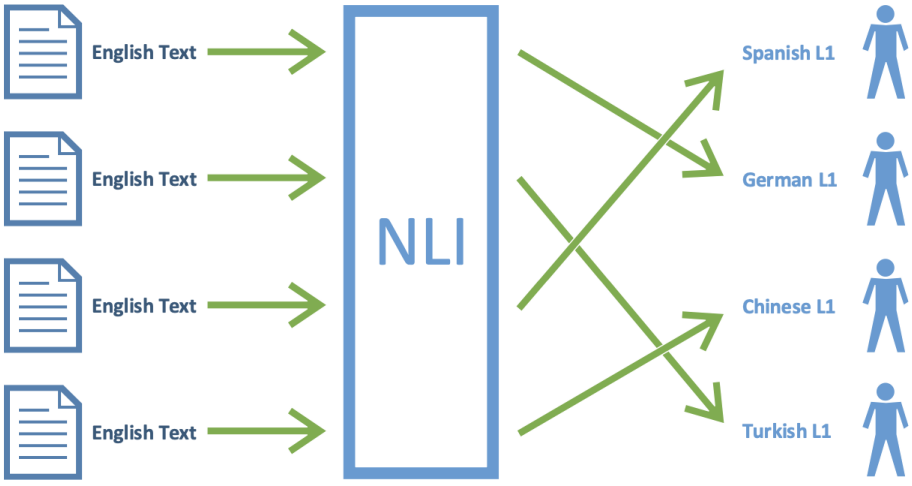
For the texts in Wikipedia we have removed all of the footnotes.

Article 3:

Approaches in this field

As we have described our task is a sub-task of NLI – Native language identification. Our review is based on the following researches by prof. Shuli winter from Haifa university and prof. moshe kopel form Bar-Ilan university.

The task of NLI is determining the native languge (L1) of an author given only text in a foreign Language (L2). We haven't been able to find any attempts in academia or in the industry to solve this task on texts in Hebrew. All the researches [4]that we have found were conducted on texts written in English.



Scheme of the main idea

[4] For example moshe kopel for bar ilan university https://www.researchgate.net/publication/221654309_Determining_an_author%27s_native_language_by_mining_a_text_for_errors

In our project we tried to use different approaches common in the industry for English texts and make an adjusted them for the Hebrew task.

Different methodologies to solve the task

Supervised approaches

The machine learning approach uses a learner which builds a classifier to recognize each category through a general inductive process by creating a set of features extracted from the documents.

1. Stylistic analysis (rule based approach) – we cast the problem as a supervised classification task and use different kinds of classification models (like logistic regression). In this approach we use different kinds of stylistic features –

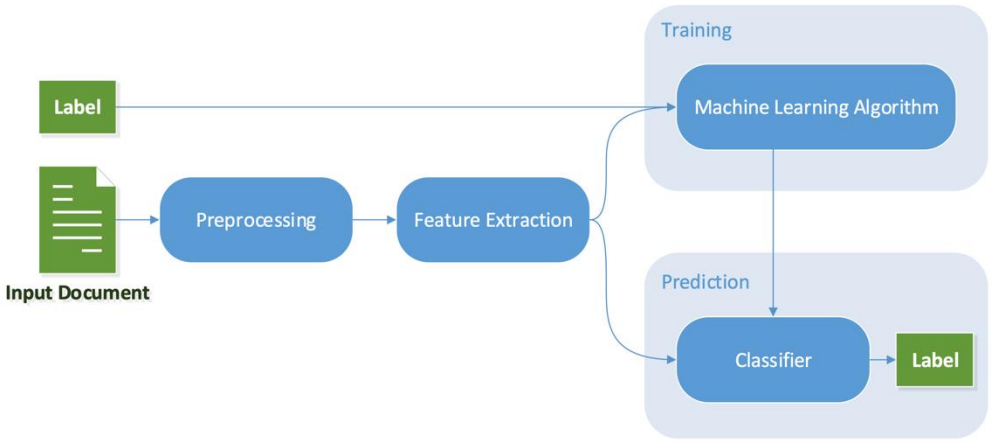
    - Function words - function words are useful for authorship attribution. Words may be useful for native langue inentification since these words are liable to be more or less frequently by native speakers in their native language. For example the word *the* is typically used less frequently by native speakers like Russia that do not have a definite article.

    - Letter n-grams – letter n-grams are very useful for this task. This includes part of speech n-grams and token n-grams. Its likely an artifact of variable usage of particular words, which is driven from the native language of the author. We also know that the distribution of different part of speech n-grams is unique for each language, so when we count the distribution of different kinds of part of speech in the text it may reflect the origin of the author.

2. An Experiment [5]made by shuly winter – they used the reddit corpus which is combined from the author meta-data and native languge (31 countries), all the speakers are fluent. In this task they used features from words that had no cultural bias and for each word they looked only on the frequencies of the words in the texts.

[5] https://www.youtube.com/watch?v=qAPYiWhNRTw&feature=youtu.be&fbclid=IwAR03IiEYeZ7489wa5idCx3Sxa2y7bCYPZ0oMMFYl5KKSiGvEV13GtDQ44zk

3. There [6]have been attempts to use only some of the features we showed before. For example, an approach using only n-grams.



A Stylistic approach – the method we used in our project

Unsupervised approach

It is done without any labelled training data, and could be considered a type of document clustering. This approach is motivated by the fact that while the collection of training documents is often straightforward, their manual labelling by domain experts is a costly and time consuming process. While some unsupervised systems may attain comparable performance to supervised classifiers on some tasks, their accuracy is generally lower.

[6] http://ceur-ws.org/Vol-2036/T4-5.pdf

Applications

This technology has practical applications in various fields. One potential application is in the field of forensic linguistics. The ability to predict the native language of the author can be a tool for authorship profiling. In order to provide evidence about the background of an author.

this tool can be useful for intelligence services and social networks, in order to find fake profiles and to point to fake news (for example a foreign government spreading false information in another language).

It could alse be used for teaching and learning a language. A rising number of language learners has led to an increasing demand for language teaching tools. In the field of second language acquisition, using this tool can help teachers to understand the difficulties and the challenging aspects of a language for learners from a specific background. It is done by identifying the L1 specific language.

Article 4

We chose a machine learning approach that uses a learner which builds a classifier to recognize each category by creating a set of features extracted from the documents and then vectorize the sentences and applying a classification model (logistic regression, naïve bayes, Random forest and even tried MLPClassifier)

In order to build the features vector we used those features:

- Function words - function words are useful for authorship attribution. Words may be useful for native langue inentification since these words are liable to be more or less frequently by native speakers in their native language. For example the word *the* is typically used less frequently by native speakers like Russia that do not have a definite article.

- Letter n-grams – letter n-grams are very useful for this task. This includes part of speech n-grams and token n-grams. In our model we used bigrams and trigrams.

- Word rank – we calculated the variance between the words in the sentence.

- Word count – we created a feature that counts the number of words because we have noticed that sentences in Arabic are often longer then sentences in Hebrew.

- Number of clauses in the sentence.

- Indicator if the sentence begins with a verb

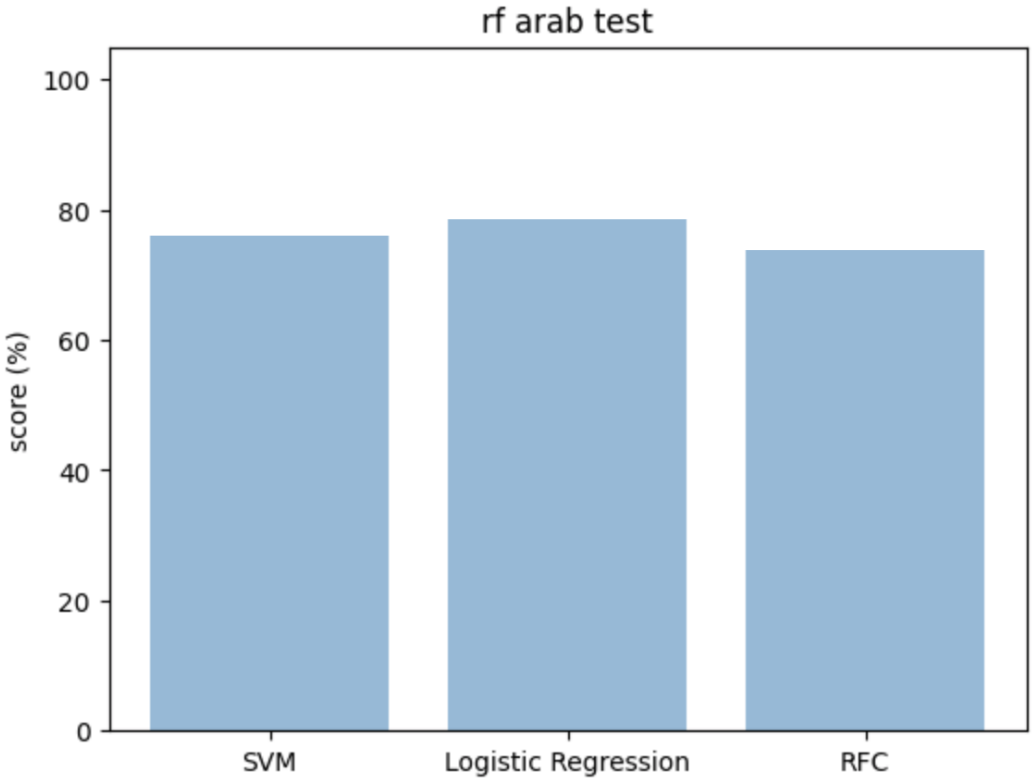
- Indicator if the sentence begins with a letter ו (in Arabic it's common to start a sentence with ו)

- Errors – we created features that reflect mistakes the may be common among arab native speakers that study Hebrew such as:

    1. Letter a instead of b (in our case – inversion between ב and פ and inversion between ק and כ.
    2. Linguistic errors – mismatch between a verb and a noun.
    3. quantity mismatch (plural/single).

After vectorising the features we made a comparison between 4 classification models (from scikit – learn library)

After training the model on the training set and evaluating on the test set we have recieved these results:



An example of the graph our model creates

We tried different classifier models and eventually we chose the logistic regression model because we noticed it's more accurate (we evaluated the test set several times and LR always got the best results).

Article 5:

Our model is a binary classification model, so, given a sentence our model can evaluate if the sentence was written by Hebrew native or an Arabic native. Our model was trained on sentences – means it's a sentence level prediction.

To remind you, our main task is classifying a text so we had to think of a way to classify a text only by a model that was trained on a separate sentences (our model was a sentences level prediction, without regard to the other sentence in the text).

Our model gives a grade: 0 for a sentence that was written by a native arabic speaker and 1 for a sentence written by native hebrew speaker. The method we used is giving a score to a text by calculating the percentage of the sentences that were classified as written by non native Hebrew speakers from all the sentences.

$$g = \frac{the\ numebr\ of\ sentences\ classified\ as\ arab\ native\ speaker}{the\ number\ of\ sentenes\ in\ the\ text}$$

$$0 \leq g \leq 1$$

$$if\ g \geq 0.5\ \ we\ classified\ the\ sentence\ as\ native\ Hebrew$$

$$if\ g < 0.5\ \ we\ classified\ the\ sentence\ as\ native\ Arabic$$

Article 6:

As we have explained in article 4, we used different classifiers that were given in scikit – learn. After training the classifier and evaluating the test set we got these results for each classifier model:

```
naive bayes classifier score:
0.7001270648030495
mlp score:
0.7407878017789072
svm score:
0.7687420584498094
lr score:
0.7662007623888183
rf score:
0.758576874205845
```

We got higher scores than we have anticipated so it is a bit of a surprise. Some of our interesting features (you can see on article 4 the other features we used in more details) are related to sentence grammar correctness in Hebrew – such as gender mismatch between adjective and subject and or quantity mismatch (plural/single). Those are features that we are certain are highly effective, since some mistakes a native Hebrew speaker would never make, but in our implementation errors can happen due to our reliance on YAP, and if it makes a mistake the model will make a mistake.

Another set of grammatical features in our model is part of speech bigrams (a one hot encoding of every possible bigram). The typical part of speech arrangement is different between any two languages, as is the case in Hebrew and Arabic. Therefore, it has proved a very effective feature.

There is also a set of semantic features in our model. We have a feature of the measure of pre-defined stop words in a sentence, and, using a list of the 50,000 most common Hebrew words, a feature of the measure of words not in the list (under the assumption that non native speakers would misspell words more often, and those mistakes would not be in the list). Using the words frequency list, there is a feature of the standard deviation of the words position in the list (not including stop words), that is meant to identify radical changes in the register of the sentence. That is under the assumptions that a typical native sentence has words in the same register, and that the higher the word's word rank the higher its register.

Article 7:

Our main change of approach was switching to sentence level prediction rather than text level prediction. After encountering a great difficulty in gathering data – specifically Hebrew written by native Arabic speakers, we chose to make the prediction in the sentence level in order to have enough training data. A technical change of approach was switching from stanza to YAP, stanza's Hebrew analysis was just not good enough for the needs of our project, despite its easier setup.

Article 8:

It appears that eventually we have achieved a high accuracy level of up to 80%, and if this is not a result of overfit, the conclusion is that there is a fundamental difference in the way non native Hebrew speakers phrase sentences than native Hebrew speakers, that our features have managed to capture.

A major improvement in our score came after introducing the n-gram POS features (from around 70% to 80%). Thus, we can deduce that POS patterns are an extremely important feature of the way a native Hebrew speaker phrases a sentence.

Furthermore, most of the features we have used are general and not specific to the native Hebrew/arabic problem, and the same applies for our evaluation method and model. Therefore, we can conclude that POS patterns, grammatical correctness, and word rank variance are universal features of the way a native language speaker uses their language.

We have reached a barrier around 80% of accuracy of sentence level prediction. Although there are surely improvements we can make to our model, we interpret that figure as an indication that often, non native Hebrew speakers phrase sentences exactly as would a native Hebrew speaker, and we conclude that there is a fundamental limit to the ability to identify an author as a native speaker or not.

Article 9:

we have some ideas we would like to further develop the project into given more data.
First of all, given more data we would like to improve the final result we got. We think that with more data we can get a better accuracy.

Second, with more data we would like to use more complex machine learning approaches using neural networks (and complicated models like Bert) and more complicated features. This methods of course require large amount of data that we don't currently possess.

More over, our current model was tested on sentence level prediction. With more data (especially texts and not separate sentences)
we can train a model for text level prediction (like we have done in the lesson for topic classification). Training the model on texts and not on a separate sentence will give us more accurate results. The main reason is that there is a connection between sentences in a text and our model doesn't consider it.

In the field of data, we believe that we can extract relevant data from academic works by non native Hebrew speakers in Israeli universities, instant messaging applications and social networks posts. The latter 2 cases are made of a large number of “short” texts, thus requiring either clever automatic data extraction techniques or manual labor.

Another possible improvement is lemmatizing our “most common Hebrew words” list; currently it can have multiple entries of the same lemma (such as אחות, אחותי, אחותך), and thus not accurately describing the word rank of the lemma. We hypothesize that the lemma carries more information regarding the word rank of the word than its conjugated form. Such a lemmatization should improve our word rank related features, which we think are some of the most significant based on the way we analyze how humans differentiate between native and non native speakers of the language. The lemmatization can be done using YAP.