Separating Guests from Topics in Podcast Episode Descriptions

Arvid Larsson
ar7144la-s@student.lu.se

Abstract

Podcasts are generally consumed by subscribing to different RSS-feeds. There could, however, be ways of generating a new podcast-feeds by picking podcast episodes from conventional feeds. This could, for example, be done by having a feed only containing episodes where a specific person is featured. A convectional search would not be sufficient to find these episodes since the name of the person could be mentioned in the metadata of the episode in the context of a Topic and not in the context of a Guest. This novel problem could be solved by using a binary classifier. In this paper, two novel solutions for the automatic annotation of training data were explored. A binary bidirectional LSTM classifier was trained on big sets of auto-annotated data and the results were compared to when the model was trained on a small set of hand-annotated data. The results show that the model trained on hand-annotated data performed better despite having vastly less training data. This indicates that quality is more important than quantity in this case. The accuracy of the model trained on hand-annotated data was 0.79.

1 Introduction

Podcasts exists as a series of episodes, usually in the form of audio files that the user can download or stream. Podcasts are usually distributed via RSS-feeds. Like classical radio, this medium often features one or several persons talking to each other. A user can subscribe to a podcast hosted by one or several persons that are of interest to the user. There are, however, many public figures that do not have their own podcast but are still featured in some podcast episodes, often as interview subjects.

A search engine could be used to generate a playlist containing episodes, picked from various podcasts, where a person of interest is mentioned in the episode title or description. However, the result from such a search would also include episodes where the person is just a topic, discussed by *other* persons. This might not be ideal for the user. A binary classifier could be used to, based on the contents of the title and description of the episode, determine whether or not a person is a Topic or a Guest and filter the episode where the person is not participating. In this text, a Guest is anyone participating in the podcast episode (including the host) whereas a Topic is a person that is mentioned in the episode description but not participating. The problem described above is illustrated in Figure 1.

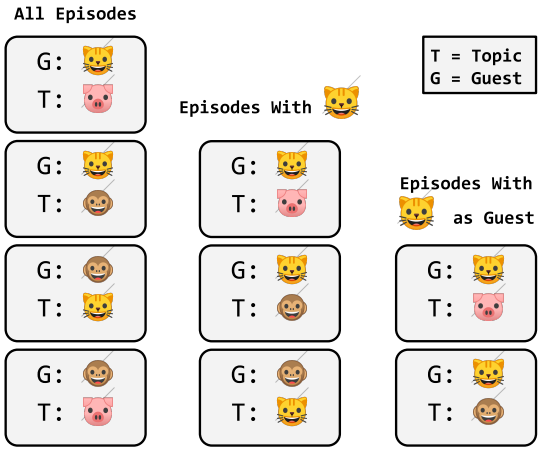

Figure 1: An example illustrating the problem presented. The column to the right contains all episodes available featuring different individuals, as Guests and as Topics. The middle column only contains episodes where a specific individual is mentioned in the episode description. The goal is to separate the episodes where a person of interest is mentioned but not participating, resulting in the third column.

2 Related work

Binary classification problems have previously been shown to be solvable using machine learning (Volkova et al., 2017). However, this novel problem does not seem to have been solved so far.

3 Data

A set of 35,000 podcast RSS-feed URLs was acquired (Telle, 2017). The podcast metadata was retrieved from using the URLs. Using a language detection library (Nakatani, 2010), all episode title and episode descriptions in English were collected resulting in a corpus of more than 1,543,558 title-description pairs.

Using the library NLKT (Bird, 2009), all personal names, (names with at least a surname and given names) were tagged. Each episode was then split into one sample for every unique name in the title-description pair, see Figure 2. By doing this, each sample contained one focus-name and zero or more non-focus-names.

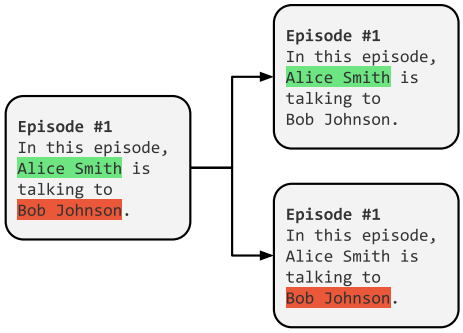

Figure 2: An example showing how every title-description pair is split into one sample for every unique name. Every sample then contains one focus-name and zero or more non-focus-names.

4 Annotation

Different annotation strategies lead to five different training sets. The first strategy was to use information from Wikidata.org for automatic labeling. The second strategy was to label data using the distribution of unique name occurrences in the corpus. The third strategy was to use a small number of hand-annotated samples to seed semi-supervised learning, this was done in two flavors. The last strategy was to only use the hand-annotated set. In all cases, the majority class was sub-sampled to create a balanced data set. It's easy to see how all strategies except the last once would lead to many mislabeled samples in the set. However, the advantage of the automatic labeling strategies is that they are far more scalable, leading to more data in total. The different sets are listed in Table 1.

4.1 Annotation using Wikidata

Wikidata is a free database facilitating data, and its connections, used on Wikipedia. By parsing the page linking to all tables containing a death date, a list of all dead people with Wikipedia articles could be obtained. In a similar fashion, a list of all people with Wikipedia articles could be assembled using the page that links to all pages listed in the category "Human".

The samples were then labeled Topics if the focus-name could be found in the list of names of dead people since dead people are unlikely to be participating in a podcast recording. If the focus-name could not be found in either of the lists, they were labeled as Guests. If the focus-name was found in the list of all people but not in the list of dead persons, the sample was discarded.

4.2 Annotation using Name Mention Distribution

The number of occurrences in the corpus for each unique personal name was counted. The distribution of these names can be seen in Figure 3. The samples were labeled with the assumption that names that occur many times in the corpus will probably belong to a public person that most time is a topic, while names with few occurrences will most likely belong to persons who more often than not is a guest. In other words, there are few people that have ever been participating in hundreds of episodes, but there are many people who have only been in a few. If the focus-name in a sample had more than 600 mentions it was labeled as a topic if it had less than 7 mentions it was labeled as a guest. The other samples were discarded.

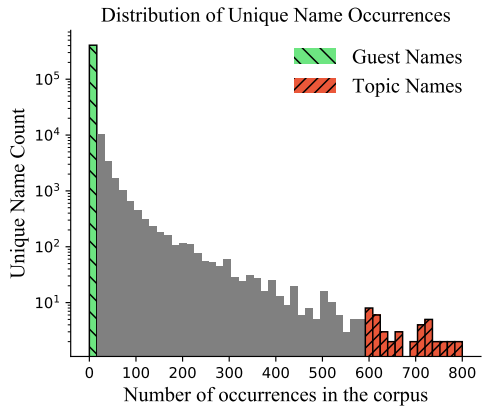

Figure 3: The distributions of how many times unique names are mentioned in the corpus. People who are mentioned a lot are unlikely to be a guests in a podcast episode.

After the labeling, all focus-names were replaced by focus-placeholders and all non-focus-names were replaced by non-focus-placeholders. This resulted in a labeled data set with 142,766 samples. The set was split 80:20 for training and validation, and 224 samples were put aside and manually annotated resulting in a test set.

5 Hand Annotation

A set of 362 samples was hand-annotated. This was used to seed a semi-supervised learning scheme. A model was trained using the hand-annotated set as training set. All unlabeled samples was then classified and the samples with the highest certainty were added to the training set, the model was then trained again using the expanded set. The number of samples to add to the training set was chosen so that the training set would expand by 10% every iteration.

The selection of classified samples were done in two flavors. In the first one (Hand-Annotated SS1) the samples with the highest confidence were selected. In the second one (Hand-Annotated SS2) random samples with more than 95% confidence were selected.

One model was also trained on the hand-annotated without any semi-supervision.

6 Preprocessing

Since the length of the samples varied from a single sentence to hundreds of words the samples were shorted using a window centered around the placeholder for the focus-name. The window used encapsulated ten words before and after the focus-name in order to capture the context around the person mentioned.

All words were encoded using the 50-dimension GloVe word-embedding (Pennington et al., 2014). All focus-names and non-focus-names in the data sets were replaced with placeholder-words. The placeholder-words for the focus-name and the non-focus-names were added to the set of vectors before the samples were encoded. This was done in order to train the model to learn the context around a name rather than the name itself.

7 Model

The model used was a bidirectional LSTM network with 128 dimensions for each direction. Dropout, recurrent dropout, and early stopping were used for regularization. One model for each of the data-sets was then trained.

A baseline model using logistic regression was trained on the samples for each annotation strategy. For the baseline model, the samples were encoded using tf-idf, instead of word-embeddings.

8 Evaluation

The models trained on the different data sets were evaluated by classifying the samples in the hand-annotated test set. Since all data sets were balanced, accuracy was used as the performance metric. The test results for the models trained on the five different data sets can be seen in Table 1. Confusion matrices for all data sets can be found in Table 2 to 5.

		Test accuracy	
Training set	**Size**	**tf-idf**	**LSTM**
Wikidata	750,806	0.65	0.62
Mention Distribution	142,766	0.49	0.53
Hand-Annotated SS1	939	0.72	0.75
Hand-Annotated SS2	939	0.72	0.73
Hand-Annotated	362	0.72	0.79

Table 1: The test accuracy and sizes for the different training set.

	Predicted	
	Topic	**Guest**
True Topic	62 %	38 %
True Guest	37 %	63 %

Table 2: Confusion matrix for the model trained on the name Wikidata set.

	Predicted	
	Topic	**Guest**
True Topic	20 %	80 %
True Guest	14 %	86 %

Table 3: Confusion matrix for the model trained on the Mention Distribution set.

	Predicted	
	Topic	**Guest**
True Topic	68 %	32 %
True Guest	36 %	64 %

Table 4: Confusion matrix for the model trained on hand-annotated set using the 1st semi-supervised learning method.

	Predicted	
	Topic	**Guest**
True Topic	76 %	24 %
True Guest	17 %	83 %

Table 5: Confusion matrix for the model trained on hand-annotated set using the 2nd semi-supervised learning method.

	Predicted	
	Topic	**Guest**
True Topic	78 %	22 %
True Guest	29 %	71 %

Table 6: Confusion matrix for the model trained on hand-annotated set without semi-supervised learning.

9 Discussion

We can, form Table 1 conclude that none of the training sets resulted in a model that performed really well. This can likely be attributed to different factors for each training set.

The strategy of using Wikidata probably lead to a high portion of mislabeled data. For example, common names are likely to be found in the list of dead people, and the samples corresponding to those names will then be labeled as Topics even though many of them are Guests.

The Mention Distribution training set might also suffer from the problem with common names. There are, however, reasons to believe that the model learned a different way of separating the samples labeled as Topics from the ones labeled as Guests in this case. If we look at the distribution of name occurrences in Figure 3, we can see that they are the majority of names are mentioned a few times. If we would then pick a sample at random we would expect to find it close to the names labeled as Guest. Since the test samples were picked at random also to find them close to the Guest names. In the confusion matrix (Table 3) we can see how almost all of the names were predicted to be guests. This leads me to believe that the learning algorithm got stuck in a different minimum than what we indented. There is likely more than one way of separating the two labels. This also leads to the question of whether or not the test set makes for a good evaluation. It's possible that random samples are not a good representation of samples that would be returned when the average user-search for a person.

Regarding the three hand-annotated methods we can see that they performed better than the other two. It's unclear weather or not the semi-supervised learning method would benefit from having more hand-annotated labels as seed.

Even if one of the models would have performed really well, the model would not lead to a good user experience if the results from a user-search would be too skewed towards either Guests or Topics. This is similar to how a diagnostic tool used to screen for a disease that has a 99 % accuracy is useless if the disease only occurs in 0.1 % of the population.

10 Future Work

It seems like the automatic labeling methods explored here are unlikely to yield positive results. More focus should be on creating a bigger hand-annotated set and optimize a model for that. When having a bigger set more effort could also be spent on exploring the semi-supervised learning schemes further.

When dealing with algorithms for automatic labeling the performance of the labeling should be thoroughly evaluated before proceeding with opti-

mizing the machine learning model.

The validity of the test scores is uncertain since the test set consisted of only 224 samples.

11 Conclusions

The problem presented seems to be solvable, but the ways of automating the data annotation explored in this paper do not seem to yield good results. Going forward, more samples should be annotated by hand.

12 Acknowledgments

I want to thank Pierre Nugges, professor at the department of computer science at LTH, for supervising me and helping me throughout the project. I also want to thank my friend, Michael Young, for the idea of custom podcast feeds that resulted in me doing this project. Thank you to Dwight Lidman and Patrik Laurell for giving me useful advice over the course of this project.

References

Edward Loper Ewan Klein Bird, Steven. 2009. Natural Language Processing with Python. O'Reilly Media Inc.

Shuyo Nakatani. 2010. Language detection library for java.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Brandon Telle. 2017. Podcast dataset. Used shows.csv, `https://data.world/brandon-telle/podcasts-dataset`.

Svitlana Volkova, Kyle Shaffer, Jin Yea Jang, and Nathan Hodas. 2017. Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 647–653, Vancouver, Canada. Association for Computational Linguistics.