Mapintel Project Report

David Silva, Prof. Fernando Bação

March 1, 2021

Abstract

Briefly summarize your previous work, goals and objectives, what you have accomplished, and future work. (100 words max) If you have a question, please use the help menu ("?") on the top bar to search for help or ask us a question.

Introduction

The Mapintel project aims at providing *Agência para o Investimento e Comércio Externo de Portugal* (AICEP) with a Competitive Intelligence (CI) tool to explore up to date articles from a myriad of national and international news sources, allowing for a new and interactive way of discovering information.

CI is concerned with gathering and analyzing information on any aspect of the business environment (competition, customers, legal framework, etc.) needed to support executives in strategic planning and decision-making. AICEP's mission is to promote Portuguese exports and to secure foreign direct investment, playing a major role in the economic development and job creation in Portugal. CI is a key component of AICEP's activity, which requires keeping track of current affairs and sift through the endless flow of news about markets, trade, industries, countries and politics.

In this project we look to answer the research question: **What aspects does a document exploration system require to extract meaningful information from a continuous flow of text documents?** We use Natural Language Processing and Machine Learning algorithms to represent and analyze the text documents, particularly we use document embedding models such as Paragraph Vector (Le and Mikolov, 2014) to represent the documents in a vector space which encodes the semantics of each document and Self-Organizing Maps (SOM) (Kohonen, 1982) to explore and visualize the different segments of documents in that space.

Literature Review

Gathering relevant information from large resource collections has been a constant need for several tasks. Information Retrieval (IR) is the process responsible for dealing with this need through efficient searching of information.

Ji et al. (2019) proposes a system for visual exploration of neural document embeddings to gain insights into the underlying embedding space and to promote the utilization in prevalent IR applications. t-SNE (Van der Maaten and Hinton, 2008) is used to project the high-dimensional data onto a 2D surface. This technique is able to capture both local and global structure from the high-dimensional data in an efficient and reliable way. In this work, the documents are embedded using the Paragraph Vector model (Le and Mikolov, 2014). The system visualizes neural document embeddings as a configurable document map and enables guidance and reasoning, facilitates to explore the neural embedding space, identifies salient neural dimensions (semantic features) per task and domain interest and supports advisable feature selection (semantic analysis) along with instant visual feedback to promote IR performance. Overall, the system provides users with insights and confidence in neural document embeddings given their black-box nature.

Lafia et al. (2019) uses SOM and Latent Dirichlet Allocation (LDA) (Blei et al., 2003) to convey the relatedness of research themes in a multidisciplinary university library. LDA is a generative probabilistic model of a corpus. The basic idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words. That said, each document is embedded in a vector space of N dimensions, corresponding to the number of topics selected. SOM

produces a landscape for exploring the topic space and provides users with an overview of the document collection and the ability to navigate (discover items of interest), change the level of detail, select individual documents and discover relationships between documents.

Kaski et al. (1998) presents the WEBSOM system - a system that organizes a textual document collection using a SOM-based graphical map display that provides an overview of the collection and facilitates interactive browsing. Kohonen (2013) revisits the topic and provides some enhancements. Here, the documents are represented with a TF-IDF weighting (Karen, 1972) and a random projection is used to reduce the dimensionality of the vector space, while preserving the similarity structure between documents. A SOM is constructed and each document is mapped into the node that best represents it. This provides exploring, searching and filtering capabilities. For example, when a node in the map is clicked, the titles of the corresponding documents and eventually some additional information such as descriptive words are presented. Also, the map is described by an automatic annotation procedure explained in Lagus and Kaski (1999), which helps to understand the semantics encoded in each map region. The user can also perform queries either using a set of keywords or a descriptive sentence. The query is then mapped into the reduced vector space and matched with the most similar documents and/or nodes. A zooming feature is also present which allows the user to explore specific regions of the map with finer detail.

Henriques et al. (2012) proposes the GeoSOM suite, a tool for geographic knowledge discovery using SOM. This tool is designed to integrate geographic information and aspatial variables in order to assist the geographic analyst's objectives and needs. The tool provides several dynamically linked views of the data consisting of a geographic map, a u-matrix, component plate plots, hit-map plots, parallel coordinate plots, boxplots and histograms. These views and their connection allows for an interactive exploration of the data.

Methods

The methodology adopted in this project can be summarized by Figure 1. First, we set up a data collection process to automatically absorb the continuous flow of news articles, then some pre-processing was applied to the data to make it usable by the embedding models and imp rove the quality of the produced document vectors. A SOM was applied to build a two-dimensional grid that is able to represent the multi-dimensional input space and therefore, the properties and relationships of the articles. Finally, a document exploration interface was built to provide the user the ability to explore and query the articles. The code developed for the project can be accessed at github.com/DavidSilva98/mapintel_project.

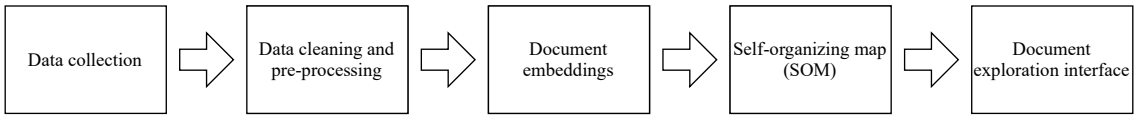

Figure 1: Methodology

Data Collection

In this project we decided to focus on how NLP and particularly sentence embeddings could help in organizing, exploring and retrieving text documents. Since the objective is to explore news articles, we used a REST API [1] to continuously retrieve English articles from multiple international sources several times a day. The API calls are performed through the AWS Lambda service [2] and the articles, as well as their metadata, are stored using the MongoDB Atlas cloud database service [3]. One particularly useful feature of the metadata is the category of the article. This can be one of the following: business, entertainment, general, health, science, sports or technology. It is important to note that the API we are using imposes some limitations that affect the data collection such as the articles being provided with 1 hour delay, having a maximum of 100 requests per day and the content of the article being truncated to 200 characters. We also developed a simple Optical Character Recognition (OCR) pipeline using the Tesseract OCR engine [4]. The purpose of this pipeline was to integrate internal documents from AICEP

[1] newsapi.org
[2] A serverless compute service that lets you run code without provisioning or managing servers
[3] A fully-managed cloud database service
[4] github.com/tesseract-ocr/tesseract

in our application, however we haven't focused on these documents so far.

Data Cleaning and Pre-processing

After loading the document corpus from the database, we concatenated the title, description and content fields in order to obtain longer and more informative documents. We proceed to clean the documents by removing non-textual patterns such as URLs and HTML tags and by removing non-English articles which are still present despite the filter applied previously. We split the document corpus into train and test set to allow for downstream unbiased performance assessments. A pre-process pipeline is applied on both corpora [5] to reduce the dimensionality of the vocabulary. The pipeline consists of removing stop words (very frequent words that are irrelevant), lowering the letter's case, removing accents and punctuation, applying stemming [6] and removing words that appear in just one document or in more than 90% of the documents.

Document Embeddings

Once the corpus is pre-processed, we encoded each document as a single vector of information. Here we used several approaches and compared them using a standardized evaluation design. As a baseline we used a Bag-of-Words (BOW) approach (Harris, 1954), which represents each document as a token histogram of the top 10000 most frequent tokens. We also used a Term Frequency - Inverse Document-Frequency (TF-IDF) approach (Karen, 1972) which tries to adjust for the fact that some words appear more frequently in general by offsetting each token frequency with the number of documents that contain the token. In these 10000 tokens, we included n-grams containing one to three words so word order information could be included in the embedding. We also used the proposed models by Le and Mikolov (2014), namely the Distributed Memory Model of Paragraph Vectors (PV-DM) and the Distributed Bag of Words version of Paragraph Vector (PV-DBOW), to learn continuous distributed fixed-length vector representations from variable-length pieces of text. These models are trained on the task of predicting words in a paragraph by looking at the context of the target word, which is encoded in the words and paragraph vectors. These vectors are the parameters of the model and are adjusted using stochastic gradient descent and backpropagation. One of the disadvantages of these methods is that to infer the embeddings of new documents, the model needs to train them which can be a problem when dealing with user queries.

Model Evaluation

To evaluate the quality of each approach, we used the corresponding embeddings and categories of each document in various tasks, similarly to some of the ones in Conneau and Kiela (2018). One of the approaches was training a logistic regression model using the embedding vectors of the train corpus to predict the article category. The accuracy of the classifier on the test corpus was used to evaluate the embeddings as the model is kept constant over the different approaches. We realized that, even though the model is kept constant, we cannot control for the interactions between each feature set and the logistic regression, which means the scores obtained don't completely isolate the embeddings performance. For this reason, a second task was proposed which consisted in classifying whether each unique test document pair belonged to the same category, based solely on their cosine similarity. The cosine similarities were converted to a range between 0 and 1 using the min-max transformation and the average binary cross-entropy over all unique pairs of documents was obtained. We also looked at the t-SNE projections of the embeddings to visualize how well they captured the semantics of the documents. Our hope was that if some semantic properties were captured, the embedding vectors would have been grouped by their categories. In the results section we will analyze how the different embedding models produced different evaluations.

Self-Organizing Map

After comparing the several embedding models, the SOM model was applied on the train embeddings of the best performing model using a fork of the SOMPY package (Moosavi et al., 2014). The grid is composed of regular hexagons as they are "visually much more illustrative and accurate, and are recommended" (Kohonen, 2013). Also, we selected the lengths of the horizontal and vertical dimensions

[5] Note: the pipeline is fitted only on the train set to avoid data leakage

[6] Term normalization process that removes the morphological and inflectional endings from words

of the grid to comply with the relation of the two largest principal components, while providing enough nodes to adequately represent the details and clusters of the input space. The oblong regular arrays have the advantage over the square ones of guaranteeing faster and safer convergence in learning. The nodes were initialized as a regular, two-dimensional sequence of vectors taken along a hyperplane spanned by the two largest principal components of the input space, providing faster ordering and convergence (Kohonen, 2001). Finally, we relied on the minimization of the quantization error (the mean distance of every data point to the corresponding best-matching unit) to select the remaining hyper-parameters of the model.

Document Exploration Interface

We extracted the fitted codebook matrix and we utilized it to build a U-matrix (Ultsch, 1993) to visualize the structures of the high-dimensional input space. By adding an interactive component to this visualization, we were able to encode several details and information within each unit of the matrix such as the distance from its neighbor units, the number of observations allocated to it and also some aggregated information from these observations. We also used the approach in Lagus and Kaski (1999) to characterize regions of the U-matrix by optimal positioning of descriptive keywords. These words function as landmarks i.e. navigational cues that help in maintaining a sense of location during the exploration of the map. Finally, we integrated the remaining components of the interface such as the search bar, a pane to preview the articles retrieved by the query and some more dynamic components to facilitate the user interaction.

Results

Present an observation (results), then explain what happened (analysis). Each paragraph should focus on one aspect of your results. In that same paragraph, you should interpret that result. In other words, there should not be two distinct paragraphs, but instead one paragraph containing one result and the interpretation and analysis of this result. Here are some guiding questions for results and analysis:

When describing your results, present your data, using the guidelines below:

- What happened? What did you find?
- Show your experimental data in a professional way. Refer to Grammar Guidelines for Reports for details on formatting. Be sure to reference figures before they appear in your paper (see Figure 2). Be sure to do the same for tables (see Table 1). For a good tool for making tables, go to tablesgenerator.com.

Figure 2: Captions go beneath figures.

Table 1: Captions go above tables.

Parameter	Symbol	Value
Residence Time	θ	90 s
Hydraulic Gradient	G	500 s^{-1}

After describing a particular result, within a paragraph, go on to connect your work to fundamental physics/chemistry/statics/fluid mechanics, or whatever field is appropriate. Analyze your results and compare with theoretical expectations; or, if you have not yet done the experiments, describe your expectations based on established knowledge. Include implications of your results. How will your results influence the design of AguaClara plants? If possible provide clear recommendations for design changes that should be adopted. Show your experimental data in a professional way using the following guidelines:

- Why did you get those results/data?
- Did these results line up with expectations?
- What went wrong?
- If the data do not support your hypothesis, is there another hypothesis that describes your new data?

Discussion

Study comparison with other studies. What were the limitations?

Conclusions

Explain what you have learned and how that influences your next steps. Why does what you discovered matter to AguaClara? Make sure that you defend your conclusions. (this is conclusions, not opinions!)

Future Work

Describe your plan of action for the next several weeks of research. Detail the next steps for this team. How can AguaClara use what you discovered for future projects? Your suggestions for challenges for future teams are most welcome. Should research in this area continue?

References

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.

Conneau, A. and Kiela, D. (2018). Senteval: An evaluation toolkit for universal sentence representations. *CoRR*, abs/1803.05449.

Harris, Z. S. (1954). Distributional structure. *WORD*, 10(2-3):146–162.

Henriques, R., Bacao, F., and Lobo, V. (2012). Exploratory geospatial data analysis using the geosom suite. *Computers, Environment and Urban Systems*, 36(3):218–232.

Ji, X., Shen, H.-W., Ritter, A., Machiraju, R., and Yen, P.-Y. (2019). Visual exploration of neural document embedding in information retrieval: Semantics and feature selection. *IEEE Transactions on Visualization and Computer Graphics*, PP:1–1.

Karen, S. J. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

Kaski, S., Honkela, T., Lagus, K., and Kohonen, T. (1998). Websom – self-organizing maps of document collections11this work was supported by the academy of finland. *Neurocomputing*, 21(1):101–117.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69.

Kohonen, T. (2001). *Software Tools for SOM*, pages 311–328. Springer Berlin Heidelberg, Berlin, Heidelberg.

Kohonen, T. (2013). Essentials of the self-organizing map. *Neural Networks*, 37:52–65. Twenty-fifth Anniversay Commemorative Issue.

Lafia, S., Last, C., and Kuhn, W. (2019). Enabling the discovery of thematically related research objects with systematic spatializations. In Timpf, S., Schlieder, C., Kattenbeck, M., Ludwig, B., and Stewart, K., editors, *14th International Conference on Spatial Information Theory, COSIT 2019*, Leibniz International Proceedings in Informatics, LIPIcs, pages 18:1–18:14. Schloss Dagstuhl- Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing. 14th International Conference on Spatial Information Theory, COSIT 2019 ; Conference date: 09-09-2019 Through 13-09-2019.

Lagus, K. and Kaski, S. (1999). Keyword selection method for characterizing text document maps. *IET Conference Proceedings*, pages 371–376(5).

Le, Q. V. and Mikolov, T. (2014). Distributed representations of sentences and documents. *CoRR*, abs/1405.4053.

Moosavi, V., Packmann, S., and Vallés, I. (2014). Sompy: A python library for self organizing map (som). GitHub.[Online]. Available: https://github.com/sevamoo/SOMPY.

Ultsch, A. (1993). Self-organizing neural networks for visualisation and classification. In Opitz, O., Lausen, B., and Klar, R., editors, *Information and Classification*, pages 307–313, Berlin, Heidelberg. Springer Berlin Heidelberg.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11).