Paper Title\* (use style: paper title)

\*Note: Sub-titles are not captured in Xplore and should not be used

line 1: 1st Given Name Surname   
line 2: *dept. name of organization   
(of Affiliation)*  
line 3: *name of organization   
(of Affiliation)*line 4: City, Country  
line 5: email address or ORCID

line 1: 2nd Given Name Surname  
line 2: *dept. name of organization   
(of Affiliation)*  
line 3: *name of organization   
(of Affiliation)*line 4: City, Country  
line 5: email address or ORCID

*This paper describe the way we decide to process data and what we use to predict information of data in the context of kaggle competition on bike sharing demand. So we describe our feature design, the methodology we use to have a good prediction and the results of our work.*

Keywords—component, formatting, style, styling, insert (key words)

# Introduction

Bike sharing systems tend to expands in many cities. However cities tries to build bike sharing systems where bike would be always available in station. Indeed, people usually commute to the same regions at the same time so some station could be empty and the service unavailable. To solve this problem, we want to move bikes at a good place and time to respond to the demand. So we think about using artificial intelligence to predict when and where regions of Montréal would be empty of bike. So we can bring bikes in this region. A kaggle competition exist on this issue and we subscribe to this competition. To use machine learning we have data on which we going to train our systems. But this data means nothing before processing and some of data are maybe useless. So in this paper we going to explain how we process training and testing data before using this processed data in a neuronal network that we are going to described.

# Data

## Selecting a Template (Heading 2)

We use the data find on kaggle. We have access to a csv file that is composed of 15 columns: date/hour, temperature (°C), drew point (°C), relative humidity (%), wind direction (10s deg), wind speed (km/h), visibility (km), visibility indicator, pressure at the station (kPa), hmdx, wind chill, weather, public holiday, station code, withdrawals and volume. All this information are on different format that we need to uniform to work on. The result of the data process should be an output able to be the input of a neural network like an array of float. We choose to use the library numpy to work with array because there are a lot of function and method implement with numpy and numpy is conform to tensorflow (the library we use for the neural network, we explain our choice of this library on the paragraph on the neural network.). In our data we have string, date, float and int. When we load data, the float are not in the good format to be process so we change it. Like we change the format of date. For the date and hour we decide to take into account the day (and after if we were in week end or not) and we divide the day in 7 time period for the hour. So we obtain as processed data of Date/Hour, a one hot vector of length 14 (then 9). There are different other feature that we transform in one hot vector: the wind direction that we divide in a one hot vector of length 4, the pressure was transform in a one hot vector of length 5. We decide to change the pressure in one vector because we think to be interest in the fact of heavy weather or not. Of course we can’t keep the station code as the way it was: we have just the station id but no information about his location and we think that the location of the station matters. So we divide Montréal in [nombre de polygone] and we do a one hot vector of length [nombre de polygone]. After we have function to know in which polygone is the station of the data row. The features "Temperature (°C)", "Drew point (°C)", "Relativite humidity (%)", "Wind speed (km/h)", "Visibility (km)", "hmdx" was normalize to have the possibility of work with this data which have different scale. We have made one function to know the percentage of non-value in a column and we note that some column are almost empty so we don’t use this column in our data mining. The columns are: [Colonne que l’on a supprimé]. The training data and the testing data need to be train in the same way: that means that even if the columns of test data are not in the same order that columns of train data, our transformed data of testing and training need to have the same header with the corresponding column. So we worked a lot with header to know where which information is. To execute the processing data, we just have to do:

data = Data(True, train\_filename)

That process the training data.

data.test\_traitement(True, test\_filename)

That process the testing data.

Of course this line of code are in the main function, as explain in the Readme, we just to need to execute model.py with the good argument to have the predicted data.

# Model

We decide to use keras from tensorflow because it is a library that all the members of team know and that’s an easy library to use. So we want to do a binary classifier. We made a Model class which implement two important things: the reduction and the classifier. After transforming our data in one hot vector, we obtain lot of feature and we decide to reduce the number of feature. So we use reduction we made a function called define\_reduction, in that way we could change the hyper parameters of the reduction and even the type of reduction (PCA or SVD) by just changing the function define\_reduction. We don’t need to change the other parts of code. A parameter of this function is the number of dimensions that we want after the reduction because PCA and SVD take this parameter as input. So it is our first hyper parameter (hyper\_param\_1). We decide to use SVD for our test thinking that using SVD or PCA doesn’t change the result. The model class contains also a function called define\_classifier\_as\_DeepNN, if we use this function, the classifier of model would be a neural network of at least 3 layers. This function take a list or an array as input parameter that determine the number of layers and the dimension of intermediate layers, it’s our second hyper parameters in case we use a neural network. To execute one model and predict the volume of our testing data we need to execute the few line of code following:

model = Model(data)

model.define\_reduction(hyper\_param\_1)

model.define\_classifier\_as\_DeepNN(hyper\_param\_2)

model.train\_model()

model.test\_model(predict\_filename)

So we can easily test our model and data processing with different hyper parameters. The data of training are really imbalanced so the accuracy is maybe not the best indicator of performance for our system. That why we use the function classification\_report which give the recall and the precision to have more information about our performance. After different test of hyper parameters we decide to use [hyper paramtre 1] as hyper\_param\_1 and [hyper\_parametre 2] as hyper\_param\_2.

# RESULT

# Futur Works

# Conclusion

#### Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

1. Table Type Styles

| Table Head | Table Column Head | | |
| --- | --- | --- | --- |
| Table column subhead | Subhead | Subhead |
| copy | More table copya |  |  |

1. Sample of a Table footnote. (*Table footnote*)
2. Example of a figure caption. (*figure caption*)

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

##### Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*
2. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
3. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
4. K. Elissa, “Title of paper if known,” unpublished.
5. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
6. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

**IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**