Project Specification

1 Formalities

1.1 Preliminary Thesis Title

Song categorization and music genre exploration on high-dimensional data using soft-subspace clustering.

1.2 Name and Email

Emil Juzovitski - emiljuz@kth.se

1.3 Supervisor @EECS

Johan Gustavsson - johangu9@kth.se

1.4 Name of the principal and name of the supervisor at the principal's workplace

- *(Principal)* Soundtrack Your Brand
- *(Supervisor)* Omar Marzouk - Omar@soundtrackyourbrand.com

1.5 Current Date

Thursday 21$^{\text{st}}$ March, 2019

2 Background and Objective

2.1 Background

This thesis is on the research topic of clustering analysis on high-dimensional data. Clustering analysis is the term used for unsupervised learning techniques that tries to group data of multiple dimensions together if they are similar.[1].

The host company is a music streaming platform company specializing in providing music for retail businesses. The platform saves the customer time by automating their music selection. The retail customer picks characteristics e.g. *energy, genre, sound* that fit their brand. Sets of songs (playlists) are automatically generated for the customer to play based on the chosen characteristics.

With the tool of clustering a belief is that similar songs can be grouped together, and, potentially uncover themes in similar songs. If a cluster is determined to have a theme, songs of that cluster can be assigned a categorical label, that is, the theme. Given enough themes, the previously unlabeled dataset becomes labeled. A resulting effect is that the now labeled data can be used for supervised learning. In addition to creating a supervised dataset, clustering songs can help playlist composers employed by the company to find candidate songs for tailored playlists reducing the time needed to find songs for a playlist.

For the purpose of this project, a dataset of songs will be used, it includes more than $5 \cdot 10^7$ data entries. The attributes of each song are metadata and an audio-embedding characterizing the song. Together they make up more than 100 attributes i.e. the dataset is of high-dimensionality.

High-dimensional datasets introduce problems when clustering. With distance measures such as the Euclidean distance, distances between objects become more similar with higher feature-dimensionality, making it harder to find the most similar cluster to an object. As such, traditional clustering algorithms struggle with finding the optimal cluster partitions.

A solution to the problem of high-dimensionality is to lower the importance of irrelevant values and increase the importance of relevant values by adding a weight to each feature. Due to the amount of features in high-dimensional data it is often hard to determine the importance of each feature manually. Feature-weighting is a technique to automate the generation of weights.

A more extensive solution to tackle high-dimensionality is to use subspace-clustering. More specifically soft-subspace clustering. In soft-subspace clustering each cluster is assigned a vector of weights for each feature. The purpose is to allow each cluster its own weight representation, as features correlate differently in different clusters. In other words: clusters have their own subspace. The subspaces reduce the amount of features that affect the distance significantly. In simplified terms this means that the dimensionality of each cluster is reduced by only allowing important features to impact the distance, resulting in better clusters.

~~As stated above, the data is mixed, i.e. the attributes are of different data types. While interval-scaled distances (numerical) can be measured well through the Euclidean distance measurement, *categorical data* cannot be measured in the same fashion without information loss. There are suitable distance measurements for *categorical data* but these measurements are inherently unsuitable for interval based data. *Generalized Gower distance(daisy function)* is one way to generalize distance measurements on mixed data. Still, traditional clustering algorithms only implement one data type distance measure, as a mixed data distance often comes with an increase of complexity on the general algorithm.~~

2.2 State-of-The-Art

While there is a rich source of papers on clustering approaches for numerical data, the same cannot be said for mixed clustering algorithms. Below is a brief summary of some mixed data approaches:

Mixed data clustering algorithms are mostly adoptions of previously existing algorithms. The perhaps most well known algorithm is a modification on *K-means* clustering called *K-prototype* [2] — it introduces a similarity measurement for mixed data . The similarity measure can be found in equation 1. w_l determines how important categorical data is compared to its numerical counterpart for cluster l. A simplification is made in the algorithm replacing local w_l for each cluster l with a single global weight w. The weight w is a user-defined input.

$$d(X_i, C_l) = \sum_{j=1}^{m_r}(x_{ij}^u - c_{lj}^u)^2 + w_l \sum_{j=1}^{m_c} \delta(x_{ij}^c, c_{lj}^c) \tag{1}$$

K-prototype has the same problems that can be assigned to *K-means*, additionally it has problems that the new similarity measure introduces, stemming mostly from the user specified weight input. State-of-the-art algorithms built on *K-prototype* try to solve a few of the problems embedded in *K-prototype* algorithm. Two fundamental problems are the requirement of k as a user input as well as the that the algorithm assumes all attributes to be of equal importance [3][4].

In [3] the *OCIL* algorithm is introduced. The problem of requiring a k — the number of clusters — as an input is solved through the use of competitive learning. A new unified and more precise similarity measure is also introduced. The argument for the similarity measure is that all numerical values are seen as one single vector requiring one weight, while categorical values require one weight each. (I will add the similarity measure later in the process, see page 4 in the cited paper)

The authors of [3] iterate on their solution in [5] taking a soft subspace approach — assigning weights to all attributes differently depending on the cluster and measuring each attributes' contribution to a cluster [6] — to allow unique cluster weights without the need of weights defined by the user while still allowing a reasonable performance. (I will expand on this much more later in the process)

A density based-approach is taken in [7] for clustering mixed data. Cluster centers are found by taking field intensity distance to account. With the now known centers, data is clustered. The algorithm is then extended for streaming data.

An ensemble method is introduced in [8] where any two clustering methods for numerical and categorical data respectively are used to to cluster the data-types separately. The clustering results are then merged on the basis that clustering results — even from numerical — are always categorical.

As the information on mixed-data is sparse it is not clear cut what clustering methods to use. A person not in the know-how could easily result in converting categorical data to numerical revoking information loss that could easily been avoided using a mixed-data technique [1]. Mixed-data techniques are not as fast as state-of-the-art numerical-clustering algorithms but make up by being compatible with mixed data. Above algorithms have capabilities to handle real-world situations where inputs are not known, and, different clusters have different attribute dependence. Jia and Cheung [5] in particular solves most if not all problems that occur when a new mixed dataset has to be clustered, while still being relatively fast.

2.3 Evaluation

In data-mining evaluation often cares about determining whether items are relevant or not [9]. In the case of clustering we want to evaluate whether resulting clusters are relevant or not.

There are two main ways of which clustering are evaluated: internal- and external-criterion [9] . In some research the categories are extended with the relative-criterion.[10]

The Internal criterion is an unsupervised validation approach and can be described as evaluating the results without respect to external information [10]. The average *silhouette coefficient* upon all datapoints $\overline{s}_{co}$ shown in 6, is one way to evaluate the internal criteria [11]. A single silhouette coefficient is shown in 5, it looks at a data points intra-cluster similarity — similarity to points within the same cluster, and, inter-cluster similarity — similarity with pointstside of the cluster. $\overline{s}_{co}$ goes between -1 and 1 where a high value (close to 1) indicates a natural clustered dataset.

When $s_{co}(X_i)$ is a high value (close to 1) a point is well clustered i.e. the intra-cluster similarity is high relative to the inter-cluster similarity [11]. The same reasoning is extended to $\overline{s}_{co}(\mathcal{D})$ where a high value is a well clustered dataset.

$$d_{avg}(X_i, C_l) = \frac{\sum_{X_k \in C_l} d(X_i, X_k)}{Count(X_k \in C_l)} \tag{2}$$

$$a(X_i) = d_{avg}(X_i, C_a) \ , \ where \ (X_i \in C_a) \tag{3}$$

$$b(X_i) = min_{C \neq C_a}(d_{avg}(X_i, C)) \tag{4}$$

$$s_{co}(X_i) = \frac{b(X_i) - a(X_i)}{max(a(X_i), b(X_i)} \tag{5}$$

$$\overline{s}_{co}(\mathcal{D}) = \frac{\sum_{i=1}^{N} s_{co}(X_i)}{N} \tag{6}$$

External criterion is supervised validation approach and can be described by the following sentence: Validation of the results by imposing a pre-defined structure on the dataset i.e. data not used for generating the clustering results [10]. As clustering analysis is an unsupervised learning method, it is often hard to assess the criteria — There is often no test data available for a new dataset. Still, to quantify how well the clustering approach works on a new dataset defining a ground truth is essential. One way is to use judges, experts in the field [9]. Given a ground truth an external measurement can be deployed.

Purity is a measurement for the external criterion. It measures the ratio of the most dominant class in a cluster. The data-points are labeled with a class beforehand on the ground truth [9]. Equation 7 defines the measurement. The equation does not penalize small clusters as such small clusters produces a high score.

$$P(W, C) = \frac{1}{N} \sum_k max_j |w_k \cap c_j| \tag{7}$$

Where:
$W = \{w_1, w_2, ..., w_k\}$ is the set of clusters, and, $C = \{c_1, c_2, ..., c_k\}$ is the set of clusters.

Another measurement is the *F-measure* shown in 10. Where P is the precision defined in 8 and R, recall is defined in 9. The measurement is a way to take account both Precision and Recall. The balanced *F-measure* is called the F_1*-measure* and is defined in 11. It weighs the impact of precision and recall the same.

$$P = \frac{TP}{TP + FP} \tag{8}$$

$$R = \frac{TP}{TP + FN} \tag{9}$$

$$F_\beta = \frac{(\beta^2 + 1) \cdot P \cdot R)}{\beta^2 \cdot P + R} \tag{10}$$

$$F_{\beta=1} = \frac{2 \cdot P \cdot R}{P + R} \tag{11}$$

Where:
TP are true positives, FP false positives, FN false negatives, and, β is a weight between P and R, a $\beta > 1$ emphasizes the R.

2.4 Interest and Objective

There are two main interests of the company:

- The company wants to find a way to efficiently create supervised training data. Clustering datapoints/songs would allow to categorize data and append an approximate label to each data point. Supervised training data would unlock the usage of supervised learning for further projects.
- Playlists are handpicked by music composers which requires extensive labour. Clustering can help to reduce the *Searching* done by the composers by giving them *candidate songs* for a cluster/themes, resulting in less time focused on irrelevant songs.

From a thesis standpoint the objective is to create a comparative analysis between specialized clustering algorithms for high dimensional data that use techniques such as feature-weighting to handle high-dimensional data and traditional clustering algorithms for the given dataset. Additionally, the objective is to create K clusters ($K > no.genres$) that can be determined to be sub-genres of preexisting genre-labels

3 Research Question and Method

How does the performance of EWKM — A soft-subspace clustering algorithm — compare to K-means — A traditional non-feature weighted clustering algorithm — on the given high-dimensional dataset?

3.1 Examination Method

3.1.1 Clustering algorithms that are candidates for a comparative analysis

W-K-means[4] A feature-weighting algorithm determining the feature weights based on inter-cluster variance. Numeric.

AWA[12] A soft-subspace algorithm determining the feature weights based on inter-cluster variance. Numeric. Breaks on zero variance of feature inter cluster subspace. FSC is an alternative that solves the problem by adding a constant in the distance function.

EWKM[13] A soft-subspace algorithm determining the feature weights based on inter-cluster variance and negative Shannon entropy. Numeric.

WOCIL[5] A soft-subspace algorithm determining the feature weights based on inter-cluster variance (through Hellinger distance) and negative Shannon entropy. Mixed.

K-means Traditional non-featureweighted algorithm.

All of the mentioned algorithms are extensions of K-means. What differs is the distance measure and the additional optimizing step of optimizing the feature weights. This optimization step requires the algorithm to go through the each data point an additional time per iteration. This allows the algorithms to still be linear in computational complexity, while minimizing the dimensionality problem.

Due to the availability of code EWKM will be compared with K-means exclusively on the audio embedding as mixed subspace methods are not available online. If there is time the report will be extended to also compare

WOCIL with K-prototypes on the whole feature-set of the data. In such case WOCIL will be implemented by the author.

3.1.2 Dataset

A dataset of *songs* will be used. It includes more than 50 million entries with each song entry holding hundreds of attributes. Attributes of a song is its metadata as well as an audio embedding — a vector characterizing the audio of the song

Not all attributes are *interval-scaled (continuous-linear, numerical).* The metadata attributes are e.g. *release year (ordinal), BPM (ordinal), explicit (binary), artist, gender (binary).*

The embedding is a vector of audio properties of the song with dimensionality in the hundreds. What each attribute means is unknown, the value of the attribute is interval-scaled.

The embedding is generated through a neural network of auto-encoders, trained on the Mel-frequency cepstrum coefficients(MFCC)[14].

To understand MFCC we need to first understand how humans perceive sound and how speech is created. Humans are more sensitive to frequency differences at lower frequencies compared to higher frequencies. The Mel-scale takes the differences in sensitivity into account by creating a logarithmic scale. Human speech is generated by the shape of the vocal tract. The shape determines what sound is created. A power spectrum describes how much power is in the frequency component of a signal. The envelope of the power spectrum can be used to represent the shape of the vocal tract of the given audio. An MFCC represents the envelope [14]. With the shape of the tract represented, generated phonemes can be determined of the vocals of the audio.

More details on how the embedding is generated is out of the scope of this thesis.

The *genre* metadata is based on the audio embedding. Making it a great validator if excluded from the dataset.

Two caveats are that certain songs can be missing one or more attributes, and, certain songs can have multiple values for one attribute e.g. *genre.*

3.2 Computation and Language

Algorithms will be implemented in C and wrapped through Python-C-API to allow the algorithm to be called in Python. This allows the performance of C while still gaining the functionality of packages such as numpy.

3.3 Expected scientific results

While we can measure the inner criterion through e.g. silhouette coefficient, a high silhouette does not necessarily translate into clusters with themes that make sense from an end-user standpoint. Instead a external evaluation — the supervised evaluation, can be used by asking the end-user about the resulting clusters and using the measurement of *purity.*

My hypothesis is that there will be a substantial clustering performance increase in terms of the inner criterion — specified as the average silhouette coefficient — compared to the k-means algorithm.

Given the ground truth to be the genres and the external criterion as purity, I expect the dominant genre on the top ten clusters based on the silhouette to be 2x as dominant in EWKM compared to K-means.

4 Evaluation and News Value

4.1 Evaluation

For the internal criterion the *average silhouette coefficient* will be used. For the external criterion the *purity* measure will be used.

In order to use any external criteria, a ground truth estimate is necessary. There are various options in this thesis on how to generate it. There are music composers available, there are 10^3 of tailored playlists, and, there is also a genre attribute that can be used as the ground truth if it is excluded from the clustering generation.

I will exclude the genre feature from the dataset. The dataset thus becomes labeled data, with the label being the genre. A basic purity test can then be done: finding the amount of songs with the dominant genre label in each cluster and dividing that with the amount of points in those clusters. In other words using the Purity measure. With a large K. a cluster with a dominant genre-label corresponds to a possible sub-genre of the dominant genre.

~~If the genre is excluded, data can get labeled on the genre. A basic purity test can then be done: finding the amount of songs with the dominant genre label in each cluster and dividing that with the amount of points in those clusters.~~

~~If we depend on the composers and want to create a purity measurement, we would have to let the composers decide if there is a theme in a cluster as well as deciding which songs are apart of that theme. To lessen the workload we could sample some clusters, and some songs of those clusters.~~

~~If we use the playlist to create a purity measurement, we would have to let the composers decide if there is a theme in a cluster as well as deciding which songs are apart of that theme. To lessen the workload we could sample some clusters, and some songs of those clusters.~~

~~The genre exclusion validation do not require any additional work, and, is less subjective. It also generates more testing data than what composers could provide. The downside of the genre exclusion option is that genres could improve the clustering results if it would be used in the clustering process. On the other hand the exclusion forces *new* themes/genres to be found.~~

The decision is to use genre exclusion with *purity* for now as it is the most robust way the project can assert the external validity. An option for the project could be to use the composers on a sample of the dataset if it was decided later that it is too favorable not to cluster on the genre.

The objective is fulfilled if we obtain the internal and external validity measurements on a clustering result and can assess a performance difference on the dataset compared to a unit conversion approach.

4.2 News Value

It is often the case in real world situations that the clustering analysis is required to be done on data fetched from a database. A database often stores data of different types and so the dataset to cluster is often of mixed data types. This thesis takes a deep-dive into state-of-the-art mixed data clustering methods and proposes a clustering solution for categorizing songs on a dataset of mixed attributes. With mixed data clustering methods not being a familiar topic for many this thesis reintroduces the clustering category with a new real world dataset to show the current performance capabilities of the field.

5 Pre-study

Focus will be on mixed data clustering algorithms that reduce the amount of user inputs, while still allowing a relatively high time-performance. Topics include: Subspace clustering and automatic K-variable detection.

5.1 Finding information

5.1.1 Initial Keywords

Clustering, Mixed data, Review, Pre-processing, soft subspace clustering, k-initialization

5.1.2 Method

- Initially start with finding Review articles on state of the art clustering methods.
- Go more in depth into algorithms suitable for dataset, look at original paper.
- Find sources on pre-processing.
- Search for mixed data solutions.

5.2 Obtaining the necessary knowledge and preliminarily references

There are numerous steps in order to go from data tostput:

- The first step is data type assertion and conversion. What kind of data is the dataset? How do we measure it can we convert it.
  - This topic is assessed by looking at how clustering is affected by data types and distance measurement.
  - Current reference is *Finding Groups In Data*(Kaufman, 1990)[1].
- The second step is to find suitable clustering algorithms. A way to start is to look at a review summarizing some of the highly used algorithms numeric of today. From there mixed data clustering algorithms will be searched for.
  - Current reference for summary is *Big Data Clustering: A Review* (Shirkhorshidi, 2014)[15].
  - Current Numerical clustering algorithm references.
    - ∗ *CLARANS: A Method for Clustering Objects for Spatial Data Mining* (R.Ng, 2002)[16],
    - ∗ *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise* (Ester, 1996)[17].

- Current Mixed attribute clustering algorithm reference.
    - *Clustering large data sets with mixed numeric and categorical values* (K-prototypes) (Huang, 1997)[2],

6 Conditions and Schedule

6.1 List of resources

- *"Playlist Composers"* - Professional playlist makers.
    - These professionals are full-time employees at the principal. It cannot be expected that they go through all songs and all clusters. However, going through samples of 10-100 top clusters is to be expected by the Music experts.
- Google Cloud
    - Cloud computing
- Machine with over 100GB of memory
    - With Google cloud any machine can be ordered up until terabytes of memory. Clusters of machines could also be ordered.
    - For the principal it is expected that the thesis will need a machine/machines that matches the requirements from my side.
- Python Clustering Library
    - Maybe pyclustering (https://github.com/annoviko/pyclustering)
    - Maybe Spark MLLib
- Apache Beam/Apache Spark
    - A distributed framework

6.2 Defined Limitations

I will describe the attributes of the dataset that are of interest for the thesis and its data types with the exception being the audio embedding. How the audio embedding is created will not be mentioned either due to company policy.

If a high performing mixed clustering method is not available, it could get excluded as an option for this thesis due to a limitation of time.

I could expand my limitations with time but for now, Constraint Clustering, Neural Networks such as SOM will not be considered at the time of writing.

6.3 Collaboration with the principal

The principal will:

- provide the dataset.
- Provide extensive support and knowledge
  - Guide me with decisions and ML problems.
  - Discussions on results
  - Might read my report but not main objective.
  - Support on how to use the company resource eco-system

6.4 Schedule

See 1 in Attachments.

References

[1] Leonard. Kaufman and Peter J. Rousseeuw. “Introduction”. In: *Finding Groups in Data*. John Wiley & Sons, Ltd, 1990. Chap. 1, pp. 1–67. ISBN: 9780470316801.

[2] Zhexue Huang. “Clustering large data sets with mixed numeric and categorical values”. In: *In The First Pacific-Asia Conference on Knowledge Discovery and Data Mining*. 1997, pp. 21–34.

[3] Yiu-ming Cheung and Hong Jia. “Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number”. In: *Pattern Recognition* 46.8 (2013), pp. 2228–2238. ISSN: 0031-3203.

[4] Joshua Zhexue Huang et al. “Automated variable weighting in k-means type clustering”. In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 5 (2005), pp. 657–668.

[5] Hong Jia and Yiu Ming Cheung. “Subspace clustering of categorical and numerical data with an unknown number of clusters”. In: *IEEE Transactions on Neural Networks and Learning Systems* 29.8 (2018), pp. 3308–3325. ISSN: 21622388.

[6] Zhaohong Deng et al. “A survey on soft subspace clustering”. In: *Information Sciences* 348 (2016), pp. 84–106. ISSN: 0020-0255.

[7] Jin-Yin Chen and Hui-Hao He. “A fast density-based data stream clustering algorithm with cluster centers self-determined for mixed data”. In: *Information Sciences* 345 (2016), pp. 271–293. ISSN: 0020-0255.

[8] Zengyou He, Xiaofei Xu, and Shengchun Deng. “Clustering Mixed Numeric and Categorical Data: {A} Cluster Ensemble Approach”. In: *CoRR* abs/cs/050 (2005). arXiv: `0509011 [cs]`.

[9] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. “Introduction to information retrieval”. In: *Natural Language Engineering* 16.1 (2010), pp. 100–103.

[10] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. “Cluster validity methods: part I”. In: *ACM SIGMOD Record* 31.2 (2002), pp. 40–45. ISSN: 0163-5808.

[11] Peter J Rousseeuw. “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis”. In: *Journal of Computational and Applied Mathematics* 20 (1987), pp. 53–65. ISSN: 0377-0427.

[12] Elaine Y Chan et al. “An optimization algorithm for clustering using weighted dissimilarity measures”. In: *Pattern Recognition* 37.5 (2004), pp. 943–952. ISSN: 0031-3203.

[13] L Jing, M K Ng, and J Z Huang. “An Entropy Weighting k-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data”. In: *IEEE Transactions on Knowledge and Data Engineering* 19.8 (Aug. 2007), pp. 1026–1041. ISSN: 1041-4347.

[14] Kuldip K Paliwal and Kaisheng Yao. “Chapter 6 - Robust Speech Recognition Under Noisy Ambient Conditions”. In: *Human-Centric Interfaces for Ambient Intelligence*. Ed. by Hamid Aghajan, Ramón López-Cózar Delgado, and Juan Carlos Augusto. Oxford: Academic Press, 2010, pp. 135–162. ISBN: 978-0-12-374708-2.

[15] Ali Seyed Shirkhorshidi et al. “Big data clustering: A review”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 8583 LNCS. PART 5. Springer, Cham, 2014, pp. 707–720. ISBN: 9783319091556. arXiv: `arXiv:1101.1881v2`.

[16] Raymond T Ng and Jiawei Han. “CLARANS : A method for clustering objects for”. In: *IEEE Transaction on Knowledge and Data Engineering* (2002).

[17] Martin Ester et al. *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*. Tech. rep. 1996.

Attachments

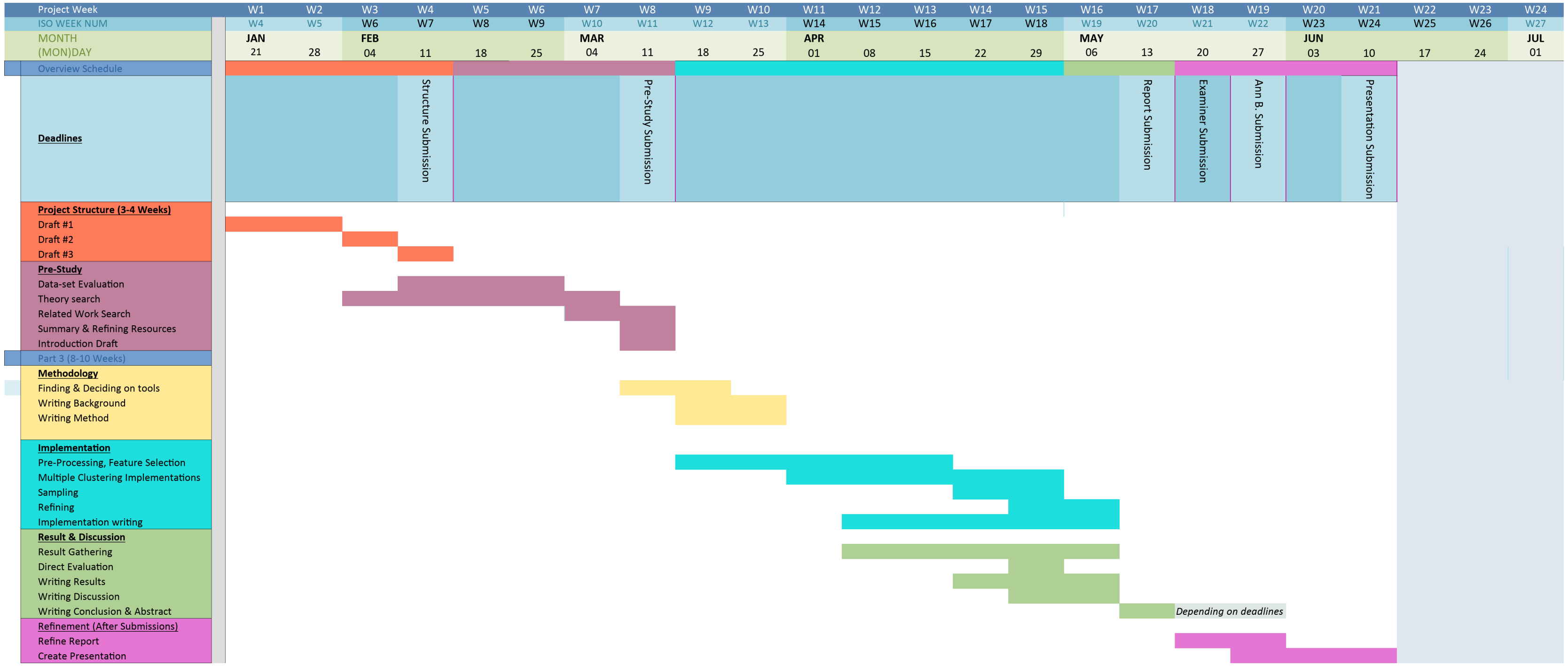

Figure 1: Preliminary Schedule