Guided Clustering for Social Media Nowcasting

Dolan Antenucci
University of Michigan
Ann Arbor, MI 48109
dol@umich.edu

The last several years have seen a growth in social media "nowcasting" applications—the use of social media data to predict real-world phenomena such as flu activity [5], unemployment behavior [1], and more. Generally these projects pick a target phenomenon that has some sort of official data (e.g. U.S. weekly UI claims), train a prediction model with this data and some features derived from a social media corpus, and finally generate a prediction with that model.

Unfortunately, there is a large class of phenomena that lack the required training data to build an accurate prediction model. For instance, the economists we worked with in 2014 [1] shared an interest the U.S. Census Bureau has with indicating human migration patterns via social media—something that is traditionally only collected every ten years. It is arguable that **most** real-world phenomena lack the training data needed to build a supervised prediction model, as traditional survey-driven data collection methods are very expensive and thus cover very few phenomena.

One important challenge of these projects is with how to pick the features derived from the social media corpus. For example, if the features are signals representing the weekly frequency of different phrases, "I lost my job" might be a good indicator for unemployment behavior. Past projects have either used data-intensive (i.e. correlation with the target) or labor-intensive (i.e. hand-filtering) processes for selecting these features. For our phenomena that lack sufficient training data, a data-intensive process is not feasible—since there is no data to test correlations against—and a labor-intensive process is still not great—for the same reasons outlined in previous work [2] (time-consuming, prone to error, etc.). Thus, choosing features for these low data phenomena is a difficult and important challenge.

One possible solution is based on clustering. The features could be clustered together based on relatedness with each other. A user would then choose a cluster that is both *interesting* and *high-quality*. An interesting cluster would be one that matches what the user believes is descriptive of the target phenomena. A high-quality cluster is one where all the elements are related with each other and which appear to identify a true real-world phenomenon.

Such a system faces two serious challenges, the first of which is scaling. A typical nowcasting feature set can be on the order of a billion or more features in size. Clustering on this scale is challenging in its own right, but supporting multiple clustering metrics—such as statistical and semantic relatedness—adds an additional complexity.

The second challenge we face is with regards to cluster quality. The user of our system has a strong bias about what he wants to see. There might be a clustering that is great according to traditional metrics, but the user may still reject it because of his biases. Consider an economist who is trying to capture a trend on unemployment: he may be expecting a seasonal spike at the start of each year due to temporary holiday jobs ending, and if he does not see the expected spike, he will reject the result. Rejecting answers due to the user's bias seems bad, but it is relatively reasonable in very low-data environments. The economist often has no data to go on, and so he must reject answers that conflict with strong domain knowledge.

We envision a system where the user can rapidly iterate over a series of clusterings. The user would be able to provide feedback on how to adjust the clusterings if he does not find his ideal answer (e.g. if "I lost my job" and "unemployment" are not in the same cluster, he user may choose to group them together). This iterative process would continue until he finds his ideal answer or eventually gives up.

A core challenge of our system will be handling the user interaction at scale. Related work in large scale consensus clustering [4] and clustering with user-feedback [3] will be a good starting point, but some sort of optimization will be needed to take the user feedback and avoid having to re-cluster the entire feature set. For example, we might be able to avoid rebuilding an entire clustering if a large subset of items is unrelated to the user's feedback.

7th Biennial Conference on Innovative Data Systems Research (CIDR '15)
January 4-7, 2015, Asilomar, California, USA.

1. REFERENCES

[1] D. Antenucci, M. Cafarella, M. C. Levenstein, C. Ré, and M. D. Shapiro. Using social media to measure labor market flows. Technical report, National Bureau of Economic Research, 2014.

[2] D. Antenucci, M. J. Cafarella, M. Levenstein, C. Ré, and M. Shapiro. Ringtail: feature selection for easier nowcasting. In *WebDB*, pages 49–54, 2013.

[3] D. Cohn, R. Caruana, and A. McCallum. Semi-supervised clustering with user feedback. *Constrained Clustering: Advances in Algorithms, Theory, and Applications*, 4(1), 2003.

[4] C. D. Ezeozue. *Large-scale consensus clustering and data ownership considerations for medical applications.* PhD thesis, Massachusetts Institute of Technology, 2013.

[5] J. Ginsberg, M. H. Mohebbi, R. Patel, L. Brammer, M. S. Smolinksi, and L. Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, February 2009.