# A Largest Common Subsequence-based Distance Measure for Classifying Player Motion Traces in Virtual Worlds

**Nikhil S. Ketkar and G. Michael Youngblood**
University of North Carolina at Charlotte
Dept. of Computer Science, College of Computing and Informatics
9201 University City Blvd., Charlotte, NC 28223-0001
{nketkar,youngbld}@uncc.edu

**Abstract**

As first-person based computer games and online virtual worlds become increasingly popular, the problem of developing classification models for player motion traces becomes increasingly important.The key applications of developing such models includes building player profiles for adaptive game play, identifying cheating in theform of bots and detecting, and curbing gold farmer activity. In this work we study the problem of developing classification models for player motion traces in virtual worlds.Specifically, we introduce a distance measure for player motion traces based on computing the largestcommon subsequence between two player motion traces. We evaluated our approach on a large corpus of player data consisting of a total of 45 binary classification problems. The kernel function based on LCS distance measure significantly out performed the Bag of Words kernel on 27 of the problems.

## Introduction

First-person based computer games and online virtual worlds are slowly replacing television and movies as popular forms of entertainment (http://www.mmogchart.com/). The distinctive feature of all first-person based computer games and online virtual worlds is that the player interacts with objects and other players in a three dimensional virtual world. Understanding and modeling human behavior in such virtual worlds becomes important as it facilitates a better game play experience. Mathematical models of human behavior can be applied to solve various problems such as building player profiles for adaptive game play, identifying cheating in the form of bots, and detecting and curbing gold farmer activity (gold farming is an activity in which a player attempts to obtain items of value that are sold to other players, usually by leveraging repetitive elements of the game mechanics).

In this paper, we study the problem of developing classification models for such and other player motion traces in virtual worlds. Intutively, this problem setting involves inducing a classification model from a set of player motion traces labeled to belong to a specific category (training data), to classify new player traces into the correct category. Specifically, we introduce a distance measure between two player motion traces that can be used as a basis for various classification models or algorithms.



## Problem Formulation

In its raw form a player motion trace $t$ consists of a sequence of points $(x_1, y_1, z_1), (x_2, y_2, z_2), ...(x_n, y_n, z_n)$. Such raw player traces can be easily preprocessed to translate the continuous three dimensional locations to a set of discrete locations. Such a procedure involves superimposing a uniform three dimensional grid with a predefined cell size $c$ over the three dimensional space and each three dimensional location maps on to a discrete location. Note that the raw data could be collected at various sampling frequencies and for very small sampling frequencies (samples acquired at intervals of less than 1 second) the player moves by a very small amount and this level of granularity is quite unnecessary. By varying the $c$ it is possible to achieve discrete sequences of locations at the required levels of granularity. Typically, $c$ is chosen to be the size of the bounding box of the player model. After such preprocessing, the player motion trace $t$ consists of a sequence of discrete locations $t = (d_1, d_2, ...d_n)$.

Given a training set of player traces categorized into two classes $\{(t_1, +1), (t_2, -1), (t_3, +1), ..\}$, where $t \in T$ (the set of all possible traces) and $\{+1, -1\}$ corresponds to two distinct classes, the problem at hand is to induce a mapping $\mathcal{F}(t) : T \rightarrow \{+1, -1\}$. Note here that we have formalized the problem as a binary classification problem and such a setting can be easily extended to multiclass classification.

## LCS-based Distance Measure

Typically, classification models are based on a table of relevant features selected by domain experts. In the case of player traces, the data cannot be easily represented as a table of features as the traces have variable length. An alternative approach is to introduce a distance measure that computes the similarity or difference between two player traces that can be used as the basis for using learning algorithms such as K-nearest neighbor or Support Vector Machines (Joachims 1999) .

A distance measure for player traces should capture the similarity between two player traces. The notion of similarity between player traces is different from the notion of similarity between strings and hence standard measures like

edit distance cannot be directly used. To define a similarity between player traces it is important to exploit the basic mechanics of game play. Typically, a player moves on the map performing multiple actions or behaviors, which leads to achieving a goal. Actions or behaviors often need to be performed in a particular sequence. An example of this would be getting to the top floor of a building and acquiring a key that can be used to unlock a door. Players may perform random actions between particular actions and behaviors, but often there there is a sequence of critical actions that allows us to uniquely identify the overall goal of the player.

For defining a distance measure between player traces, our basic idea is to use the notion of largest common subsequence (LCS) (Hirschberg 1977) to measure the similarity between two traces. The intuition behind using LCS is to account for fragments of similarity between two traces. For example, suppose that some traces consist of an important set of behaviors that are sequentially repeated in each of the traces. However, these repeating behaviors are interlaced with other actions which are not common to all the traces. The uncommon actions, in a sense represent the variability or the noise in the data and should be ignored. What should be considered are the sequentially repeating, common aspects of the traces, which are in fact captured by the LCS. In order to take this into consideration, we define the similarity measure as follows.

$$D(X,Y) = \frac{(LCS(X,Y))^2}{Length(X) \cdot Length(Y)} \quad (1)$$

where X and Y are the two traces under consideration. Note that this similarity measure is typically a number between 0 and 1. Identical traces will have a similarity measure of 1 while completely dissimilar traces will have a similarity measure of 0. The LCS problem is quite well studied in literature with numerous applications in bioinformatics. An O(mn) time algorithm (where m and n are the lengths of the input sequences) for LCS can be found in (Wagner and Fischer 1974).

## Experimental Evaluation

The central question in our experimental evaluation is how well the LCS-based distance measure performs while classifying player traces. To this end we conduct extensive experimentation with the transfer learning dataset introduced in (Cook, Holder, and Youngblood 2007). This dataset consists of human player traces at a number of levels, scenarios, and player types. We work with a subset of this dataset and consider player traces in level 6, which consists of relatively long and complicated sequences of tasks. This particular level consists of a total of 5 scenarios each of which has 2 types. This leads to a total of 10 classes, and 45 binary classification problems (classifying between 2 pairs of classes of all possible pairs).

For each of the 45 binary classification problems we compare the leave-one-out error achieved by a support vector machine using the LCS based distance measure of the kernel, a support vector machine using a bag of words kernel, and a classifier that predicts the majority class as a baseline. Note that a bag of words kernel basically computes the dot product of the vectors representing the counts of a particular location and is quite commonly used in text classification. We compare the LCS-based distance measure against the bag of words kernel as it allows us to test if the LCS measure does in fact capture important common subsequences. The bag of words kernel does not consider the sequences but only the presence and the counts of locations in a trace. If the LCS-based distance measure does outperform the bag of words kernel then there is a clear indication that player traces do contain distinguishing common subsequences that characterize player behaviors.

The initial raw data (of all the traces in the 10 classes) consisting of a series of locations in three dimensional coordinate space was preprocessed by superimposing a grid equal to the bounding box of the player character. Out of the 45 binary classification problems, the LCS-based approach outperformed the bag of words kernel on 27 problems. Based on our experimental evaluation, we can conclude that the LCS-based distance measure captures important subsequences of player actions and behaviors which can be used to classify the intention or goal of the player. It is also important to note that in a few cases the LCS-based distance measure performs poorly as compared to the bag of words kernel. Our conjecture is that in these cases the LCS-based approach is essentially overfitting the data, that is, identifying a common subsequence which has occurred only by chance.

## Conclusions and Future Work

We introduced a longest common subsequence based similarity measure for classifying player motion traces. Promising experimental evaluation on a large corpus of player trace data also indicated that player traces often do contain subsequences of behaviors and actions that can identified by the distance-based similarity measure and leveraged for classification. Experimental results also indicate that in certain cases, the LCS-based distance measure overfits the data by identifying common subsequences that have occurred only by chance. We plan to further investigate this issue as a part of our future work.

## References


Cook, D. J.; Holder, L. B.; and Youngblood, G. M. 2007. Graph-based analysis of human transfer learning using a game testbed. *IEEE Trans. Knowl. Data Eng.* 19(11):1465–1478.

Hirschberg, D. 1977. Algorithms for the longest common subsequence problem. *Journal of the ACM (JACM)* 24(4):664–675.

Joachims, T. 1999. Making large-scale support vector machine learning practical, Advances in kernel methods: support vector learning.

Wagner, R., and Fischer, M. 1974. The string-to-string correction problem. *Journal of the ACM (JACM)* 21(1):168–173.