Match Cutting: Finding Cuts with Smooth Visual Transitions Supplementary material

Boris Chen Amir Ziai Rebecca S. Tucker Yuchen Xie
{bchen, aziai, btucker, yxie}@netflix.com
Netflix Inc.
Los Gatos, CA, USA

1 Annotation task details and dataset statistics

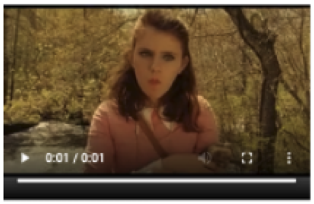
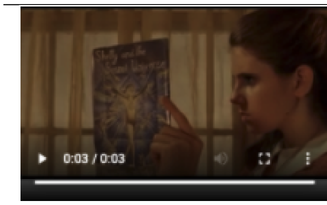
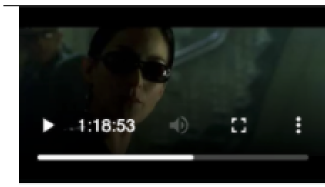
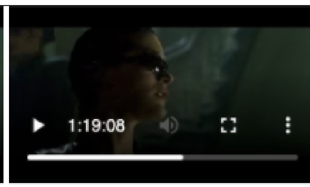
In this section we describe the rules developed with our in-house editors for the annotation of match cuts, including examples of match cuts that violate or follow the rules. We show the user interface for annotation, then provide some additional data set statistics.

1.1 Character frame match cutting

1.1.1 Rules

1. Proportions and scales of the characters should be the same.
2. Character poses should be similar.
3. Shots of the same person are okay, as long as there is something different about the shots. E.g. different location, clothes, time of day.
4. Shots should not be too similar.
5. Matches should be between characters, not objects.

1.1.2 Examples

Example		Match?	Rules violated
		No	1
		Yes	None
		No	2, 3, 4

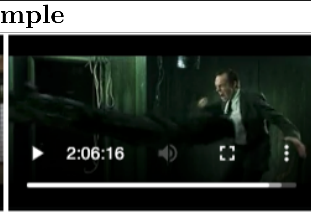
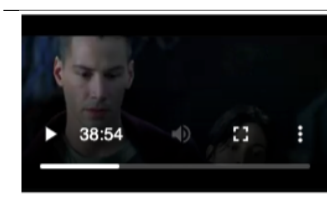
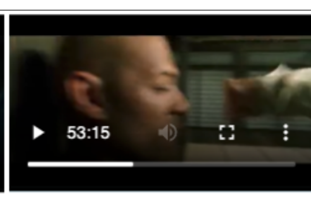
Examples are from Moonrise Kingdom (2012) [4] and The Matrix (1999) [71].

1.2 Motion Match Cutting

1.2.1 Rules

1. Characters/objects should be moving the same way or the camera motion should be similar. E.g. the camera moves the same direction, or an action-reaction pair in which they move opposite directions.
2. Number of subjects does not have to be the same, as long as the movement, pace and direction are similar.
3. Shots should not be blurry even if the motion is matching.

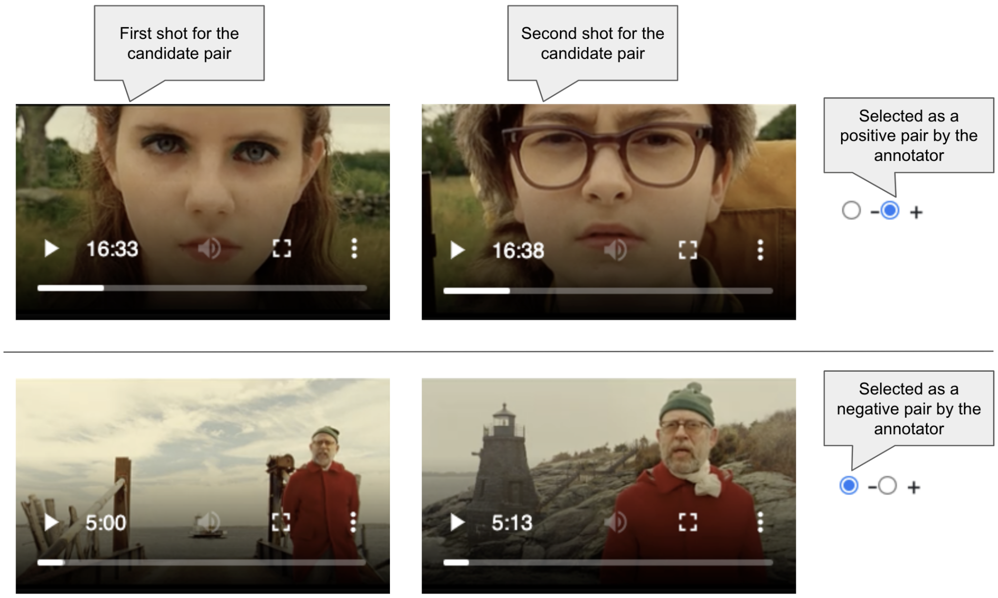
1.2.2 Examples

Example		Match?	Rules violated
		Yes	None
		No	3

Examples in this table are from The Matrix (1999) [71].

1.3 User interface for annotation

We built a custom application for presenting pairs of shots to annotators and collecting labels.



Examples are from Moonrise Kingdom (2012) [4].

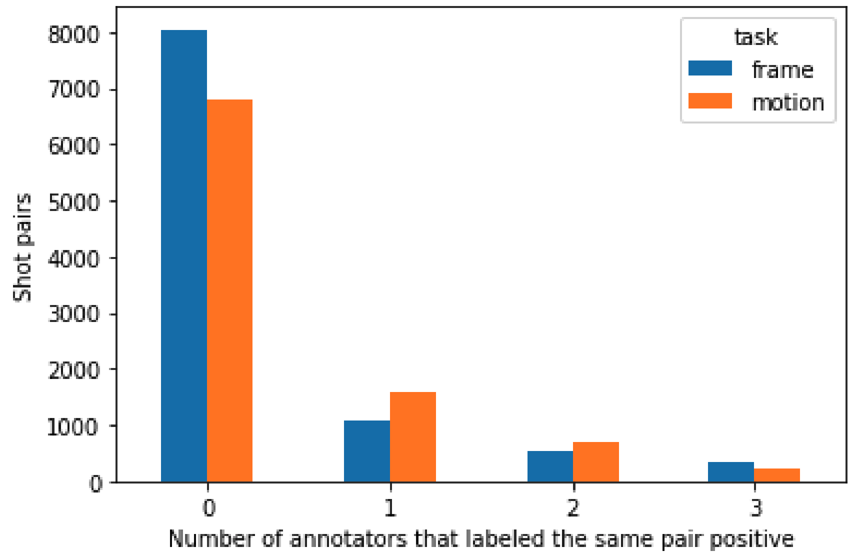
1.4 Dataset statistics

Task	Frame	Motion	Overall
Annotated pairs	9,985	9,320	19,305
Positive pairs (majority label)	867	927	1,794
Positive rate	0.087	0.099	0.093
Pairs with perfect agreement	8,373	7,027	15,400
Perfect agreement rate	0.839	0.754	0.798

1.5 Heuristic positive rate

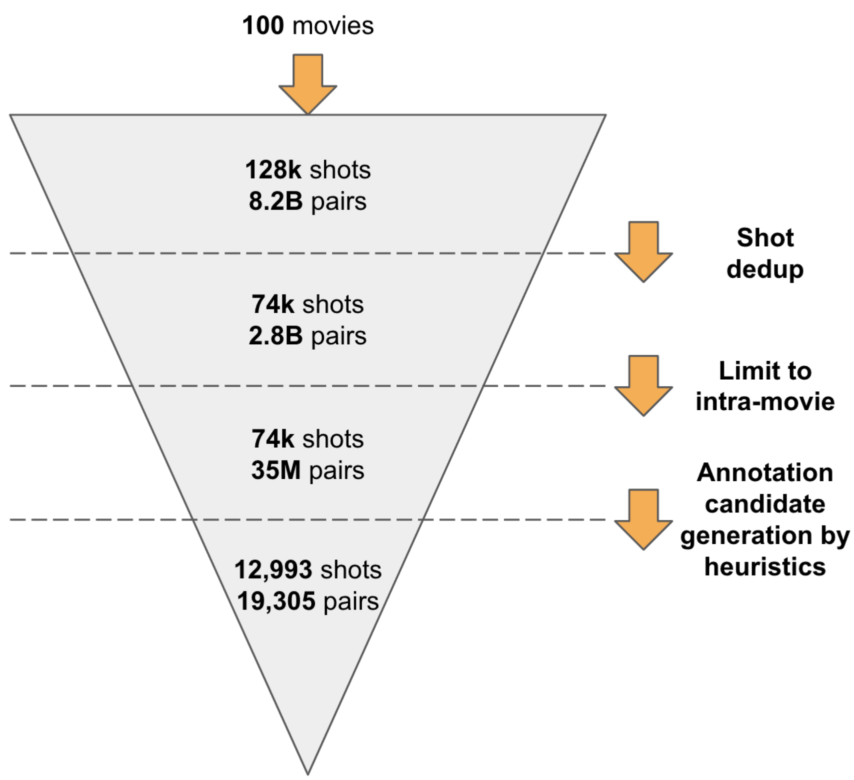
Heuristic	Pairs selected	Positive pairs	Positive rate
h_1	5,000	69	0.012
h_2	5,000	808	0.161
h_4	5,000	543	0.109
h_5	5,000	494	0.099

1.6 Annotator-level agreement by task



1.7 Annotation candidate pair generation

1.7.1 High-level process



1.7.2 Statistics

	After shot segmentation	After dedup	Limit to intra-movie	Annotated
Shots	128,202	74,493	74,493	12,993
Shot pairs	8,217,812,301	2,774,566,278	34,554,612	19,305

2 Title set and shot statistics for the released dataset

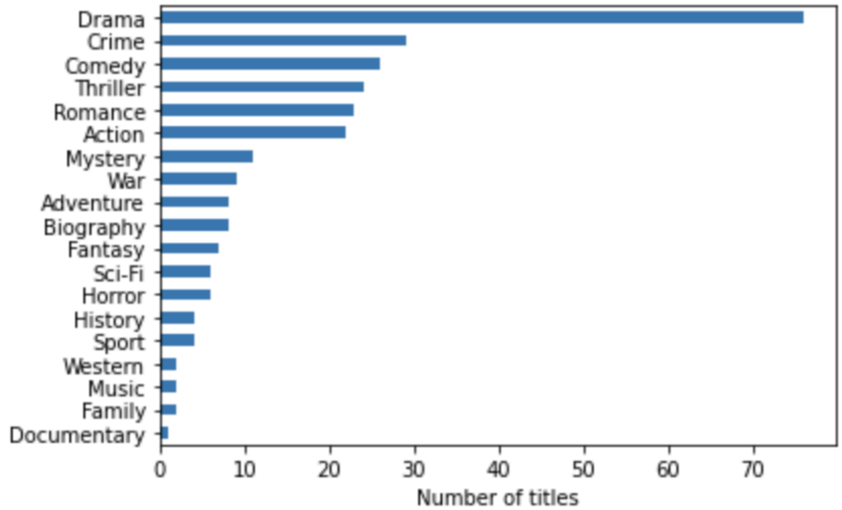
2.1 Titles

IMDB ID	Title	Country
tt0050706	Mon Oncle (1958)	France
tt0059592	Pierrot le Fou (1965)	France
tt0061722	The Graduate (1967)	USA
tt0061781	The Firemen's Ball (1967)	Czechoslovakia
tt0066921	A Clockwork Orange (1971)	UK
tt0070245	Hiroshima Death Match (1973)	Japan
tt0070246	Battles Without Honor and Humanity (1973)	Japan
tt0071315	Chinatown (1974)	USA
tt0079182	Vengeance Is Mine (1979)	Japan
tt0080610	The Last Metro (1980)	France
tt0081505	The Shining (1980)	UK
tt0090257	My Sweet Little Village (1985)	Czechoslovakia
tt0092099	Top Gun (1986)	USA
tt0092603	Babette's Feast (1987)	Denmark
tt0095250	The Big Blue (1988)	France
tt0095765	Cinema Paradiso (1988)	Italy
tt0099685	Goodfellas (1990)	USA
tt0101700	Delicatessen (1991)	France
tt0106332	Farewell My Concubine (1993)	China
tt0108289	Flirting Scholar (1993)	Hong Kong
tt0108656	Crime Story (1993)	Hong Kong
tt0110201	Hail the Judge (1994)	Hong Kong
tt0111797	Eat Drink Man Woman (1994)	Taiwan
tt0112769	La Cérémonie (1995)	France
tt0114369	Se7en (1995)	USA
tt0118749	Boogie Nights (1997)	USA
tt0118799	Life Is Beautiful (1997)	Italy
tt0118845	Happy Together (1997)	Hong Kong
tt0133093	The Matrix (1999)	USA
tt0175880	Magnolia (1999)	USA
tt0178868	Ringu (1998)	Japan
tt0190332	Crouching Tiger, Hidden Dragon (2000)	Taiwan
tt0208092	Snatch (2000)	UK
tt0250494	Legally Blonde (2001)	USA
tt0266697	Kill Bill: Vol. 1 (2003)	USA
tt0308476	The Cuckoo (2002)	Russia
tt0338013	Eternal Sunshine of the Spotless Mind (2004)	USA
tt0373074	Kung Fu Hustle (2004)	Hong Kong
tt0378194	Kill Bill: Vol. 2 (2004)	USA
tt0385004	House of Flying Daggers (2004)	China
tt0387898	Caché (2005)	France
tt0407887	The Departed (2006)	USA
tt0427954	The Protector (2005)	Thailand
tt0443706	Zodiac (2007)	USA
tt0457430	Pan's Labyrinth (2006)	Mexico
tt0468565	Tsotsi (2005)	UK
tt0469494	There Will Be Blood (2007)	USA
tt0477348	No Country for Old Men (2007)	USA

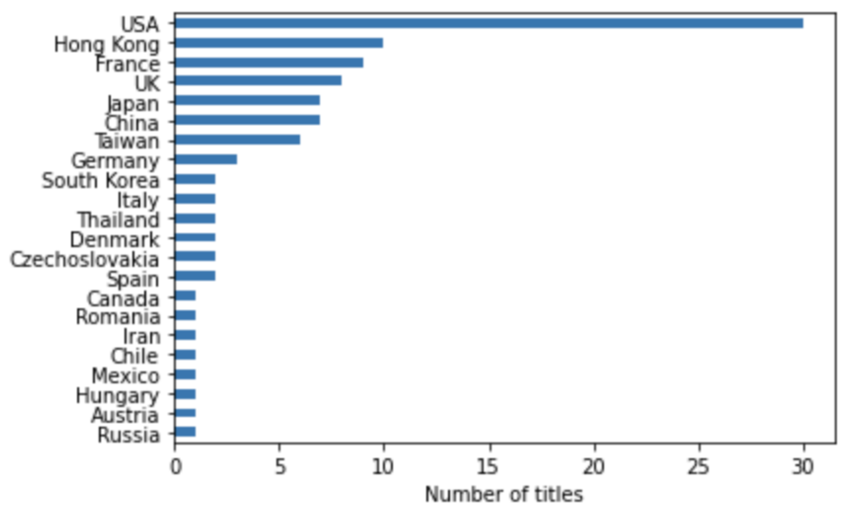
tt0765128	Oceans (2009)	France
tt0780504	Drive (2011)	USA
tt0810819	The Danish Girl (2015)	UK
tt0844347	Midnight Sun (2006)	Japan
tt0887883	Burn After Reading (2008)	USA
tt0913425	Broken Embraces (2009)	Spain
tt0940709	CJ7 (2008)	Hong Kong
tt0947798	Black Swan (2010)	USA
tt0993846	The Wolf of Wall Street (2013)	USA
tt1063669	The Wave (2008)	Germany
tt1220719	Ip Man (2008)	Hong Kong
tt1232829	21 Jump Street (2012)	USA
tt1255953	Incendies (2010)	Canada
tt1276104	Looper (2012)	USA
tt1386932	Ip Man 2 (2010)	Hong Kong
tt1462900	The Grandmaster (2013)	Hong Kong
tt1504320	The King's Speech (2010)	UK
tt1533117	Let the Bullets Fly (2010)	China
tt1560747	The Master (2012)	USA
tt1568346	The Girl with the Dragon Tattoo (2011)	USA
tt1602620	Amour (2012)	Austria
tt1611840	Once a Gangster (2010)	Hong Kong
tt1649443	[REC] 4: Apocalypse (2014)	Spain
tt1748122	Moonrise Kingdom (2012)	USA
tt1800241	American Hustle (2013)	USA
tt1832382	A Separation (2011)	Iran
tt1853728	Django Unchained (2012)	USA
tt1974419	The Neon Demon (2016)	Denmark
tt2059255	No (2012)	Chile
tt2070649	Silenced (2011)	South Korea
tt2084970	The Imitation Game (2014)	USA
tt2115388	Love is Not Blind (2011)	China
tt2258281	Beyond the Hills (2012)	Romania
tt2267998	Gone Girl (2014)	USA
tt2488496	Star Wars: Episode VII - The Force Awakens (2015)	USA
tt3421514	Supercondriaque (2014)	France
tt3501416	Assassination (2015)	South Korea
tt3508840	The Assassin (2015)	Taiwan
tt3672840	Dragon Blade (2015)	China
tt3700392	Heidi (2015)	Germany
tt3808342	Son of Saul (2015)	Hungary
tt4176826	Look Who's Back (2015)	Germany
tt4273292	Under the Shadow (2016)	UK
tt4967094	Our Times (2015)	Taiwan
tt5576318	Who Killed Cock Robin? (2017)	Taiwan
tt5580036	I, Tonya (2017)	UK
tt5593416	Peach Girl (2017)	Japan
tt5827496	At Cafe 6 (2016)	Taiwan
tt5866930	The Adventurers (2017)	China
tt6157626	Legend of the Demon Cat (2017)	China
tt6298600	The Miracles of the Namiya General Store (2017)	Japan
tt6788942	Bad Genius (2017)	Thailand

2.2 Genre breakdown

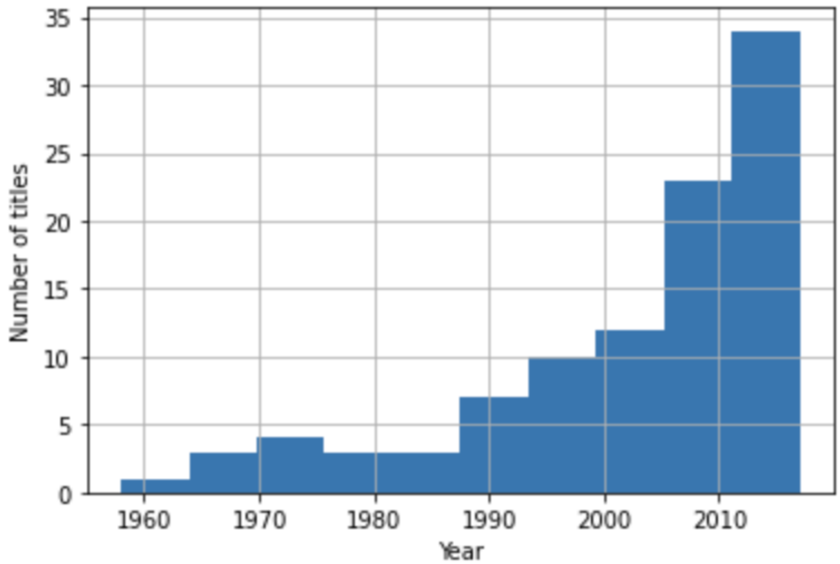
Note that titles can have more than one genre.



2.3 Country breakdown



2.4 Release year



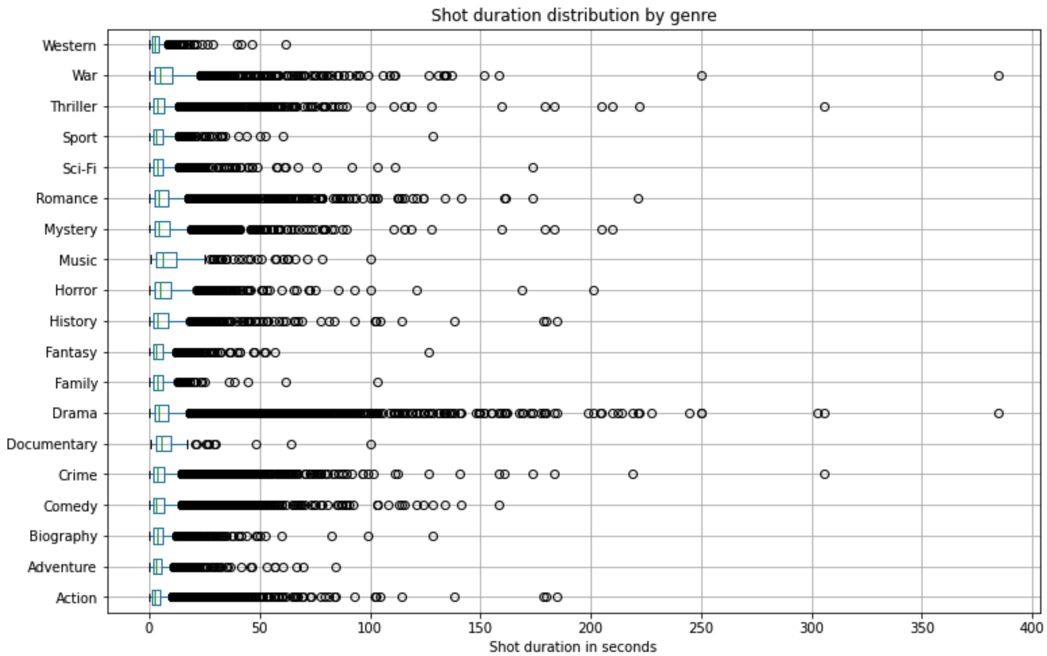
2.5 Shot duration statistics

The duration values are in seconds. Note that these values are computed for the subset of shots that we are releasing (not the entire set of shots in all the titles that we have considered).

Count	**Mean**	**Std**	**Min**	**25%**	**50%**	**75%**	**Max**
21,205	8.174	15.136	0.240	2.083	3.879	8.091	384.500

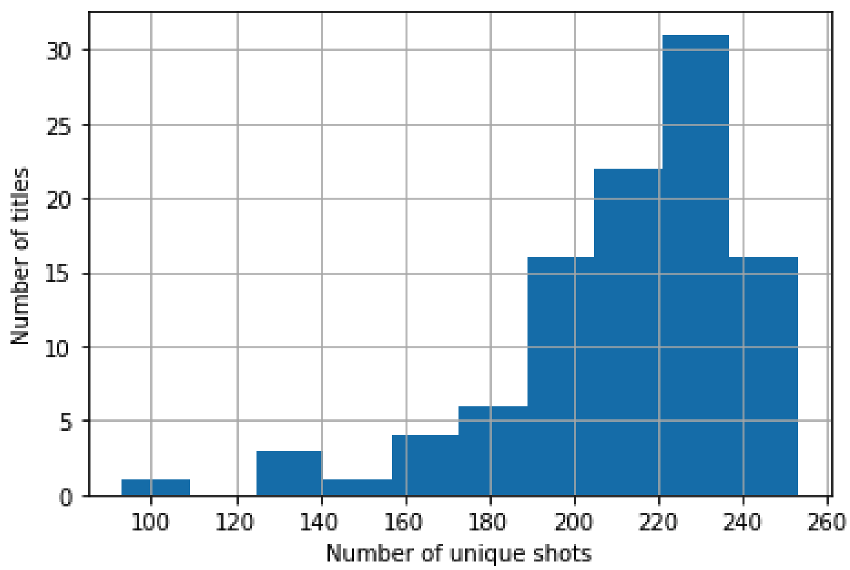
2.6 Shot duration distribution by genre

Note that these values are computed for the subset of shots that we are releasing (not the entire set of shots in all the titles that we have considered).



2.7 Number of unique shots by title

Note that these values are computed for the subset of shots that we are releasing (not the entire set of shots in all the titles that we have considered).

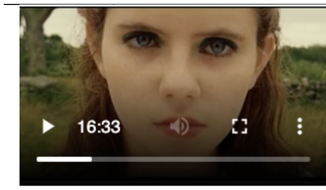
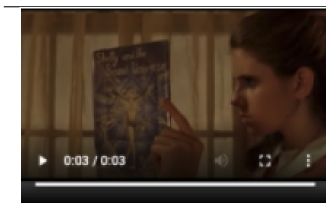
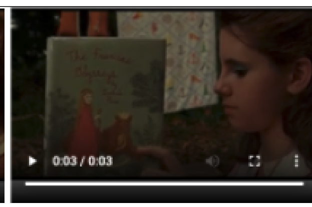


3 Evaluation

3.1 Average Precision (AP)

For match cutting, we surface a ranked list of pairs to editors. Ideally, the best candidates should be placed at the top of this list. Average Precision (AP) is an information retrieval metric that captures this setup. AP ranges between 0 and 1, where a higher value reflects a higher quality of retrieval.

To demonstrate how AP is calculated in our context, consider the following toy dataset with three labeled pairs (all pairs are from Moonrise Kingdom (2012) [4]):

Pair		**Match?**	**ID**
		Yes	A
		No	B
		Yes	C

$AP = 1$ is achieved when scores for the positives pairs (i.e. A and C), are higher than the score for the negative pair. For instance, if the scores are 0.9, 0.1, and 0.8 for A, B, and C respectively, then we have $AP = 1$. (In this case, the list above would be reordered as A, C, B before it was presented to the editors.)

AP drops below 1 as the scores cause more negatives to be interleaved with positives. For instance, if the scores are 0.9, 0.8, and 0.7 for A, B, and C respectively, then we have $AP = 0.83$.

We use the implementation provided by `scikit-learn` [56]. The following Python snippet shows how AP is calculated for these two cases:

```python
from sklearn.metrics import average_precision_score as ap

# after sorting by score we compute precision at each depth
# if the instance is positive and then divide by the number of positives
assert ap(y_true=[True, False, True], y_score=[0.9, 0.1, 0.8]) == (1 + 1) / 2
assert ap(y_true=[True, False, True], y_score=[0.9, 0.8, 0.7]) == (1 + 2 / 3) / 2
```

3.2 Baseline

Unlike some metrics such as the Area Under the Receiver Operating Characteristic curve (AUROC), AP is not agnostic to the prevalence of the positive examples (we will call this p). In other words, we can expect $AUROC = 0.5$ for random guessing regardless of the value of p, but $AP = p$ (in expectation) if scores are randomly generated.

Since match cutting is a novel task and no open source benchmarks exist, we treat the positive prevalence p as our baseline, and expect our system to achieve $AP > p$.

The following Python snippet demonstrates that the expected value of AP is p:

```python
import numpy as np
from sklearn.metrics import average_precision_score as ap


def random_ap(n: int, p: float) -> float:
    """
    n is the number of candidates.
    p is the positive prevalence.
    """
    assert 0 < p < 1
    scores = np.random.rand(n)
    pos = int(round(p * n))
    true = [True] * pos + [False] * (n - pos)
    return ap(true, scores)


def ap_mean(n: int, p: float, rounds: int, precision: int = 2) -> None:
    aps = [
        random_ap(n=n, p=p)
        for _ in range(rounds)
    ]
    return round(np.mean(aps), precision)


assert ap_mean(n=10_000, p=0.2, rounds=1_000) == 0.2
assert ap_mean(n=10_000, p=0.8, rounds=1_000) == 0.8
```

3.3 Heuristics

All heuristics described in section 3.3 produce a score given a pair of shots, which can be used for evaluation as described in the previous section. These scores can be used in the same way that we use the output score of a classification model. The only difference is that unlike learned models that can be trained with different seeds, there's no similar source of variation for heuristics. Therefore we only report a single value instead of mean and standard deviation.

4 Experiment 2 hyperparameters

For all experiments we used:

- `TripletMarginMiner` with `type_of_triplets="hard"`
- training batch size of 256

For character frame we use 128 and 1024 hidden units for the first and second layer respectively, and for motion we use 256 and 1024 hidden units in the first two layers of MLP. For character frame we used 300 epochs and for motion we used 100.

4.1 Tuning ranges of hyperparameters

For both tasks we used the following tuning ranges:

Hyperparameter	Range
`temperature`	$[10^{-3}, 1]$
`learning_rate`	$[10^{-4}, 10^{-1}]$
`weight_decay`	$[10^{-4}, 10^{-1}]$

4.2 Tuned hyperparameters

Hyperparameter	Character frame	Motion
`temperature`	7.362×10^{-3}	1.3412×10^{-2}
`learning_rate`	3.147×10^{-3}	4.056×10^{-4}
`weight_decay`	10^{-4}	4.54×10^{-4}